

The Waite Group


PLUME
WAITE

INTRODUCING THE TRS-80[®] MODEL 100

EVERYTHING YOU NEED TO GET
STARTED ON THE MODEL 100
PORTABLE COMPUTER

- Simple, jargon-free explanations
- Explains the built-in software:
TEXT, ADDRSS, SCHEDL, BASIC,
TELCOM
- Takes you from beginner to
advanced use
- Covers the new optional disk
drive and video interface

by
Diane Burns
and S. Venit



Another book by the bestselling authors of *ASSEMBLY LANGUAGE PRIMER FOR THE IBM® PC & XT* and *BLUE-BOOK OF ASSEMBLY ROUTINES FOR THE IBM® PC & XT*. . . . And rave reviews for The Waite Group:

"An outstanding example of how to write a technical book for the beginner . . . refreshingly enjoyable . . . accurate, readable, understandable, and indispensable. Don't stay home without it."

— Ken Barber, reviewing *CP/M Primer*,
in *Microcomputing*

"Mitch Waite . . . has left a distinctive contribution to the literature of computer graphics . . . seeing it here is like understanding it for the first time."

— *Computer Graphics Primer*, reviewed in
Computer Graphics World

"It's hard to imagine that a field only a decade old already has a classic, but Waite's book is just that."

— Tony Dirksen, reviewing *Computer Graphics Primer* in
Interface Age

"... an outstanding reference work, aside from its obvious benefit as an instructional text . . . superb writing style . . ."

— Chuck Dougherty, reviewing *Soul of CP/M* in *Cider*

"The demystification of a complex technological development . . . rating: 100."

— *8086/8088 16-bit Microprocessor Primer*, reviewed in
The Reader's Guide to Microcomputer Books

"[This book] can have anybody writing and understanding assembly language programs within one hour."

— Alan Neibauer, reviewing *Soul of CP/M*,
in *80 Microcomputing*

"All-inclusive, beautifully organized, easy-to-use reference that helps you wrest order from chaos . . ."

— *CP/M Bible*, reviewed in *Byte Book Club Bulletin*



Diane Burns

Diane Burns left her desk on Wall Street to become an active consultant to businesses in the San Francisco area, specializing in office automation and telecommunications. Ms. Burns is coauthor of *The SuperCalc Primer* and *Practical Finance on the TRS-80® Model 100*. Her articles on microcomputer software appear regularly in national magazines. Her professional commitment is to produce informative books which are easy to read.



S. Venit

S. Venit earned her Master's Degree in Economics from Georgetown University. She is coauthor of *The SuperCalc Primer* and *Practical Finance on the TRS-80® Model 100*, and designer of the *Star Forces* series of self-guided educational materials for young science students. Ms. Venit is an active board member for several non-profit associations, including Bay Area Career Women and Women Entrepreneurs. In touch with the perceptions of new computer owners, her books and articles are grounded in practical applications which she has developed in hands-on computer training classes through local retail stores and education departments.

INTRODUCING

THE TRS-80[®]

MODEL 100

by Diane Burns
and S. Venit

WITHDRAWN



A Plume/Waite Book
New American Library
New York and Scarborough, Ontario

NAL BOOKS ARE AVAILABLE AT QUANTITY DISCOUNTS WHEN USED TO PROMOTE PRODUCTS OR SERVICES. FOR INFORMATION PLEASE WRITE TO PREMIUM MARKETING DIVISION, NEW AMERICAN LIBRARY, 1633 BROADWAY, NEW YORK, NEW YORK 10019.

Copyright © 1984 by The Waite Group, Inc. All rights reserved.
For information address New American Library.

Several trademarks and/or service marks appear in this book. The companies listed below are the owners of the trademarks and/or service marks following their names.

Apple Computer, Inc.: Apple

Atari Inc.: ATARI

CompuServe Inc., an H&R Block Company: CompuServe, InfoPlex

Dow Jones & Company, Inc.: Dow Jones News/Retrieval

Epson America, Inc.: Epson MX-80, MX-100, FX-80

International Business Machines Corporation: IBM, IBM PC

Tandy Corporation: TRS-80 Model 100 Portable Computer, Radio Shack



PLUME TRADEMARK REG. U.S. PAT. OFF. AND FOREIGN COUNTRIES
REGISTERED TRADEMARK — MARCA REGISTRADA
HECHO EN WESTFORD, MASS., U.S.A.

SIGNET, SIGNET CLASSIC, MENTOR, PLUME, MERIDIAN and NAL BOOKS are published *in the United States* by New American Library, 1633 Broadway, New York, New York 10019, *in Canada* by The New American Library of Canada Limited, 81 Mack Avenue, Scarborough, Ontario M1L1M8

Cover design by Michael Manwaring

Illustrations by Stuart Bradford

Technical Illustrations by Winston and Karen Sin

Typography by Walker Graphics

First Printing, November, 1984

1 2 3 4 5 6 7 8 9

PRINTED IN THE UNITED STATES OF AMERICA

Contents

Acknowledgments viii

Introduction	1
1 Quick Computerese	7
Another Language? 7	
The Secret Code 8	
2 Getting Started	13
Taking a First Look 13	
The Power Sources 15	
The Screen 16	
Le Menu 16	
The Keyboard 17	
Setting the Date and Time 19	
A Note About File Size 21	
Moving On 21	
3 A Terrific TEXT Editor	23
A Quick Memo 24	
Working with SELECTed Text 30	
Using the FIND Function 34	
Global Search and Replace 35	
4 Getting Organized	37
Creating NOTE.DO 38	
Using SCHEDL 40	
Managing NOTE.DO 43	
Other Kinds of Schedules 43	
5 More than an ADDRSS Book	47
The ADRS.DO File 48	
Using the ADDRSS Program 49	
Using TELCOM with the ADRS.DO File 50	
More Complex ADRS.DO Files 53	
A BASIC Mailing Label Program 55	
A BASIC Program to Build Your Address File 58	
A Simple ADRS.DO File for Printing Mailing Labels 60	

6	<i>BASIC Basics</i>	63
	Creating a Program 64	
	Storing a Program 68	
	Running a Program 69	
	Editing a Program 71	
	Renaming a File 72	
	Deleting a File 72	
	On Your Own 73	
7	<i>Printing Files</i>	75
	What You Need 76	
	Hookup to a Printer 76	
	Printing the Screen Image: PRINT Key 79	
	Printing Whole Text Files: CTRL Y 79	
	Printing BASIC Programs: LLIST 80	
	A BASIC Print Program: LPRINT 81	
	Printing in SCHEDL and ADDRSS : F4 82	
8	<i>Storing Files on Cassette</i>	85
	Watch Your Bottom Line 85	
	What You Need 86	
	How to Hook up Your Model 100 to a Cassette Recorder 86	
	Some Special Tips 86	
	Saving and Loading Text Files 88	
	Saving and Loading BASIC Files 92	
	Summary 97	
9	<i>TELCOM and the Marvelous Modem</i>	99
	The New World 99	
	What You Need 101	
	Modem Hookup Direct to Phone Line (DIR) 101	
	Modem Hookup Through Acoustic Coupler 103	
	Getting Started 104	
10	<i>Window on the World — Information Networks</i>	113
	Information Networks 114	
	What You Need 114	
	Using CompuServe 115	
	Using Dow Jones 125	
	Coding an Auto-Log-On Sequence 125	
	On Your Own 127	
11	<i>Hookup to Other Computers: The RS-232 Port</i>	129
	What You Need 130	
	Hookup 131	
	Saving and Loading Text Files 131	
	Saving and Loading BASIC Files 137	
	Summary 139	

12	<i>Using the Optional Disk/Video Interface</i>	141
	What is the Disk/Video Interface? 141	
	Connecting the Disk/Video Interface to the Model 100 142	
	Starting Up the System 147	
	Getting Organized 149	
	Backing Up the System Diskette 152	
	Using TEXT with the Disk/Video Interface 154	
	Using BASIC with the Disk/Video Interface 156	
	Summary 157	
13	<i>The Bar Code Reader — More Versatile than You Think</i>	159
	What Are Bar Codes? 159	
	What's So Great About a Bar Code Reader? 161	
	What You Need 162	
	Setting Up Your Bar Code Reader 165	
	Operating Your Bar Code Reader System 168	
	Using the Inventory Programs 170	
A	<i>BASIC Error Codes</i>	174
B	<i>ASCII Character Codes</i>	176
	<i>Index</i>	179

Acknowledgments

We would like to acknowledge several people without whose help this book could not have been completed: Robert Lafore, Lyn Cordell, Winston and Karen Sin, Lisa Yount, Harry Henderson.

To Our Readers



Introduction

The Radio Shack TRS-80® Model 100 has launched a revolution in the world of personal computers. Before the advent of this exciting new machine, a computer this small could not do useful work, and one large enough for serious computing could not be carried and used on-the-go with any degree of convenience.

The Model 100 has changed this picture dramatically. It is truly portable, small enough to slip into your briefcase and light enough that you can balance it on your lap. Because it will run for many hours on its own internal batteries, you can use it virtually anywhere: in buses, limousines, and airplanes; in your hotel room, on construction sites, or even while backpacking!

Who Should Read This Book

This book is aimed at anyone who owns a TRS-80® Model 100 or is thinking of buying one and wants to know if it will meet his or her needs. It will serve equally well whether you are an experienced computer user or a novice who has never used a computer before. We have tried to explain things clearly and without jargon, and at the same time to be brief and succinct so that you can learn what you need to know as quickly as possible.

In the first chapter, we give the novice a solid grounding in computer terms and concepts. Then we introduce the Model 100 and go on to tell you everything you need to know to get the most out of its powerful features. We cover all the built-in software and all the peripherals, such as printers and the new disk drive and video interface.

Before we get down to these details, however, let's review the major features of the Model 100 and see why it's so different from other computers.

What Makes the Model 100 So Remarkable?

The most obvious feature of the Model 100 is its small size. It measures less than 9 by 12 by 2 inches and weighs less than four pounds! By contrast, other computers that are somewhat optimistically called "portable" by their manufacturers weigh in at a hefty 20 pounds or more. You might carry one of these large computers with you, but you are unlikely to actually put it on

your lap and use it on an airplane or in a car. The Model 100 is truly *portable*, not just *transportable*.

Several technological advances make the small size of the Model 100 possible. First, it uses a *CMOS microprocessor chip*, which requires much less power than ordinary computers. This enables the Model 100 to run off batteries instead of requiring a wall plug and bulky power supply. Second, the Model 100's viewing screen is a *liquid-crystal display* (LCD) rather than the large video tube used in almost all other computers. The LCD is small, flat, and light; without it the Model 100 could not exist in its present form.

A less obvious but nevertheless revolutionary feature of the Model 100 is its *built-in software*. Most computers require that you load programs from external devices such as a cassette or disk drive. The Model 100 has five of the most useful programs built in, so you never need to load them. These are the text editor, TEXT; the scheduling and address list programs, SCHEDL and ADDRSS; the telecommunications program, TELCOM; and the BASIC language interpreter. Later, we'll describe all these programs in detail.

In addition to the Model 100's built-in software, you can also load your own programs into memory and keep them there as long as you want. Even if you turn the computer off and put it away for a week, your programs will still be there when you turn it on! Again, most other computers require disk- or cassette-storage devices to save and load programs. By eliminating these bulky and inconvenient peripherals, the Model 100 ensures that you need carry only the computer itself: one small flat box in a neat black carrying case contains everything you need.

Another major feature of the Model 100 is its built-in modem, an electronic device that translates computer data into a form that can be transmitted over phone lines. Modems for other computers are usually separate devices, costing hundreds of dollars, and software for the modem also costs extra on most computers. With the built-in modem and TELCOM program, you can use the Model 100 to send and receive information over the phone lines — and you can do this anywhere there is a phone, whether it's a hotel room in San Francisco or a bar in a Portuguese fishing village.

Finally, the Model 100 is a well-built, professional computer. The screen is not as large as on some machines, but it is large enough for most computing tasks, and the Model 100's portability more than makes up for the modest screen size. The keyboard is completely professional: full size, with a nice feel, so that touch typists will have no trouble going full speed. In fact, many users of larger computers actually prefer the feel of the Model 100 keyboard to their other machines.

What's It Good For?

Because of its unique capabilities, the Model 100 is being used in entirely new ways by people who never thought computers would be practical for their needs.

In Colorado, several engineers at a petroleum company carry Model 100s to record and interpret drilling data. Because the Model 100 is scarcely larger than a notebook, taking it into the field is no inconvenience.

In a major hospital in New York City, several doctors take the Model 100 on their rounds to update patient files and schedule appointments. Now, they say, it doesn't matter that no one can read their handwriting!

A loan officer at a bank in California uses the Model 100 to file important loan data on his clients. Now, instead of digging through files an hour before an appointment, he takes his Model 100 with him and reviews pertinent information moments before paying a visit to a major client.

In Florida, a magazine reporter takes the Model 100 on interviews, to press conferences, and even on vacations. When her story is complete, she finds a phone and sends the story back to the home office, using the Model 100's built-in modem.

The owner of a typesetting firm in Oregon uses the Model 100 to take notes of telephone conversations, client meetings, and appointments. His desk and pants pockets used to be full of little slips of paper. Now, this vast array of miscellaneous notes is available at his fingertips, and individual items can be searched for and retrieved almost instantly.

What's in This Book?

In this book we cover all the important aspects of operating the Model 100, from unpacking it and turning it on, to using the sophisticated TEL-COM and TEXT programs, to hooking the Model 100 up to other computers. We'll present a brief synopsis here, so you'll know what to expect.

Chapter 1, "Quick Computerese", is written for the computer novice and provides the background for understanding the rest of the book, indeed, for understanding personal computing in general. Experienced computer users can probably skip this chapter without harm.

In Chapter 2, we take you on a guided tour of the Model 100. You'll learn how to set it up and get it going and see what all its plugs and switches do. We explain the various power sources and how they work, and even how to set the time and date so that the Model 100 is ready to do your bidding.

In Chapter 3, we describe what is probably the most used feature of the Model 100: the built-in TEXT program. This is a surprisingly sophisticated

text-editing program that permits you to type letters, memos, articles, and reports on your Model 100 and then edit them until they are exactly to your liking. TEXT has cut-and-paste capabilities and the same cursor movement controls found in full-scale word-processing programs. The documents you create with TEXT can be printed out or transferred to other computers. We'll show you all these techniques.

In Chapters 4 and 5, we cover the built-in programs SCHEDL and ADDRSS. These are mini-database programs that you can use to help organize your life. SCHEDL takes the place of your pocket calendar. With it you can record all your appointments, including the time and date, the people you're meeting, the purpose of your visit, and any other relevant information. The big advantage SCHEDL has over a written notebook is that you can use the computer to "search" through your schedule to find a particular name, date, or other key word and then instantly display this information. It's like having a very efficient, lightning-fast secretary in your briefcase. ADDRSS performs a similar function for your address book. By putting your address book into the Model 100, you give yourself the capability to search for and instantly recall any name, address, phone number, or other pertinent information. You can even use the Model 100 to automatically dial a particular person's phone number!

In Chapter 6, we explain the use of the BASIC program interpreter built into the Model 100. BASIC allows you to run programs that have already been written or to create your own programs. We do not teach you how to program in BASIC; that is beyond the scope of this book. For a complete introduction to programming in BASIC, see *Mastering BASIC on the TRS-80® Model 100*, by Bernd Enders (New York: Plume/Waite, New American Library, 1984).

What you *will* learn in this chapter is how to load existing programs from cassette or type them in directly, as well as how to save them in memory as files and execute them. This knowledge takes you beyond the programs built into the Model 100 so that it becomes a general-purpose computer, as versatile as any of its larger brothers.

Once you've entered a report or other document into the Model 100, you may want to print it out. You can connect the Model 100 to a printer to produce a hard-copy version of what's in the machine. In Chapter 7, we show you how to do this. You'll also learn how to print out an image of exactly what appears on the screen and how to print out listings of BASIC programs.

In Chapter 8, we tell you all about using a cassette tape recorder to save and load programs and data. This is important to know when the memory gets too full to hold everything you want to put in it.

In Chapter 9, we explain how to use the Model 100 to communicate over the phone lines, using the built-in modem and TELCOM program. This leads into Chapter 10, where we show you in detail how to use your telephone and the Model 100 to access the exciting new database services CompuServe® and Dow Jones News/Retrieval®. Get news and stock quotes, do electronic banking, or go shopping without ever leaving home!

In Chapter 11, we explain how to connect the Model 100 to other computers. When you've mastered this technique, you can think of your Model 100 as a peripheral of your larger computer — or perhaps the other way around! You'll find this chapter especially useful if you use your Model 100 for editing text or for gathering information at a remote site that you want to bring home and upload into your larger computer for more sophisticated processing.

In Chapter 12, we describe the optional disk drive and video interface. This new peripheral permits you to save all your files and programs to a standard 5¼-inch diskette, a much faster system than using cassettes. It also lets you use a standard TV set or computer monitor to view the computer's output, rather than using the LCD screen. In effect, this peripheral transforms your Model 100 into a capable desktop machine, with most of the advantages of a full-size computer. Finally, in Chapter 13, we tell you how to use the optional bar code reader and its software.

We hope you'll enjoy this book and find it a useful introduction to the TRS-80® Model 100. We've tried to distill our hundreds of hours of experience on this machine into these chapters, including practical hints and tips to make using the machine fun and easy for you.



1

Quick Computereze

Many people are confused by the terms used in the world of computers, so we'll begin by defining some of these terms and showing you their practical applications. If you're already familiar with the basic "lingo", feel free to skip this chapter. But if the sentence "Hook up the RS-232 cable to the serial port with the null modem adaptor" seems like gibberish to you, then read on.

One day, a fellow visited Mexico with a Spanish-speaking friend. Returning to the hotel after a sumptuous afternoon meal, he realized he'd left his hat at the restaurant. His friend helped him formulate the appropriate question and then sent him off to retrieve the hat. With great confidence, he said (in Spanish), "I left my hat here. Did you find it?" Unfortunately, he had no way of understanding the waiter's reply, which was spoken rather quickly and seemed, for a moment, endless. (They finally reverted to sign language, and he found his hat on the coat rack.)

Learning to use a computer may often give you the same feeling: even when you think you know what to tell the computer, you may not get the answer you expected, or one you can understand! The vocabulary associated with computers is not quite as foreign as Spanish might be to some of us, but it is necessary to understand some basic words. The words may be common English words that have taken on a new meaning in "computereze", or they may be terms that occur only when referring to computers.

Another Language?

Every activity has its own vocabulary, and working with computers is no exception. A weaver passes the *weft-bearing shuttle* through the *shed* of warp threads held apart by the *hettle*. A sailor calls, "Prepare to *come about*", the crew releases the *starboard jib sheet*, and everyone ducks and scrambles as the *boom* flies overhead. So, it's not surprising to find that a *hacker* hooks up a *modem* to dial up a *mainframe* and *download* a file.

As soon as you add a computer to your list of tools, you'll need to open up a new dictionary of terms to help you communicate with the machine itself, the suppliers, the consultants, and the know-it-all next door (who got *his* computer last week). The best way to learn this (or any) new language is to practice it, that is, to *do* something with the computer. Don't worry about memorizing all the terms explained in this chapter. As you continue to *do* things with your Model 100, you will learn the words most pertinent to your own activities. Most of us can name the parts of a car engine that the mechanic has mentioned most often, even if we don't know exactly how they work.

The Secret Code

A *computer* is a machine that can "remember" and "manipulate" binary numbers. A *binary number* is usually written as a series of ones and zeros (01011010, and so on). In the computer, these values are *stored* as positive or negative charges (or *bits*) on a wired *memory chip* (which is something you may never need to see or talk about, unless you want to add or replace chips in your own computer).

Because one bit is simply a 1 or a 0, bits are usually stored in groups of eight called *bytes*. Everything you type into the computer is stored in bytes. Here are some examples of one-byte values:

01000001 (This is the value given to the letter A)

01100001 (This is how a lowercase *a* is stored)

00000000 (This is the lowest possible eight-bit binary value)

11111111 (This is the highest possible eight-bit binary value)

A computer's size is often measured by how many memory chips it contains or how many individual bytes of information it can hold in *memory*. The standard Model 100 comes with about 8,000 (8k) bytes of memory. Of these, 3,000 (3k) bytes are used by the computer itself and the built-in *programs*, BASIC, TEXT, TELCOM, ADDRSS, and SCHEDL. Therefore, when you first turn on your Model 100, the bottom line on the screen of an 8-k machine shows that you have about 5,000 bytes free. You can add more memory chips to upgrade your Model 100 to 16k, 24k, or a maximum of 32k. Your Radio Shack dealer must do this, because it involves delving into the computer's innards.

The memory in the Model 100 is divided into two parts: *ROM* (Read Only Memory) and *RAM* (Random Access Memory). In ROM the built-in operating system (which controls many routine functions such as displaying

the main menu, and finding or executing programs) is stored. ROM cannot be changed (except by the manufacturer). RAM is where all of *your* work, including programs and text files, is recorded. On most machines, the RAM information is lost as soon as the power is turned off, so, in order to *save* the information, files are regularly copied onto a magnetic *disk* or *tape*. The Model 100 is one of the first computers with an internal nicad battery, giving it a “permanent” RAM memory. You can turn the machine off without losing the information in RAM, so you don’t need to copy the files from RAM to disk or tape.

Information is stored in the Model 100 memory under *named files*. As you add new files to the memory, the number of free memory bytes decreases. When the amount of available memory becomes too small, you will want to copy files to a cassette tape or disk so that you can clear space in your machine by deleting files.

A file can be a typed memo, a BASIC program, a list of names and addresses and so forth, and a filename can be any word up to six letters long. In addition to the name you give the file, the Model 100 adds a two-letter suffix or *extension*, preceded by a period, to identify the file as a BASIC program (.BA) or a TEXT file (.DO).



The Model 100 *screen* is your “window” into the memory. The screen shows you what you are entering as you type, and gives you messages or prompts you for information. On the main menu, the *cursor* highlights a particular filename, so that the filename shows in *reverse video*. In a text file or in BASIC, the cursor is a blinking square that shows you where the next typed character will appear on the screen. *Arrow keys* help you *scroll* through the file and cause the eight-line screen display to change. We’ll show you how to do this in Chapter 2.

The Model 100 has several *ports*, or plug openings, that allow you to connect various devices to the machine. It can be hooked up to a *cassette tape* recorder, to a larger computer, or to a *disk-storage* device. As we mentioned earlier, you will probably want to copy old files from memory onto a tape or

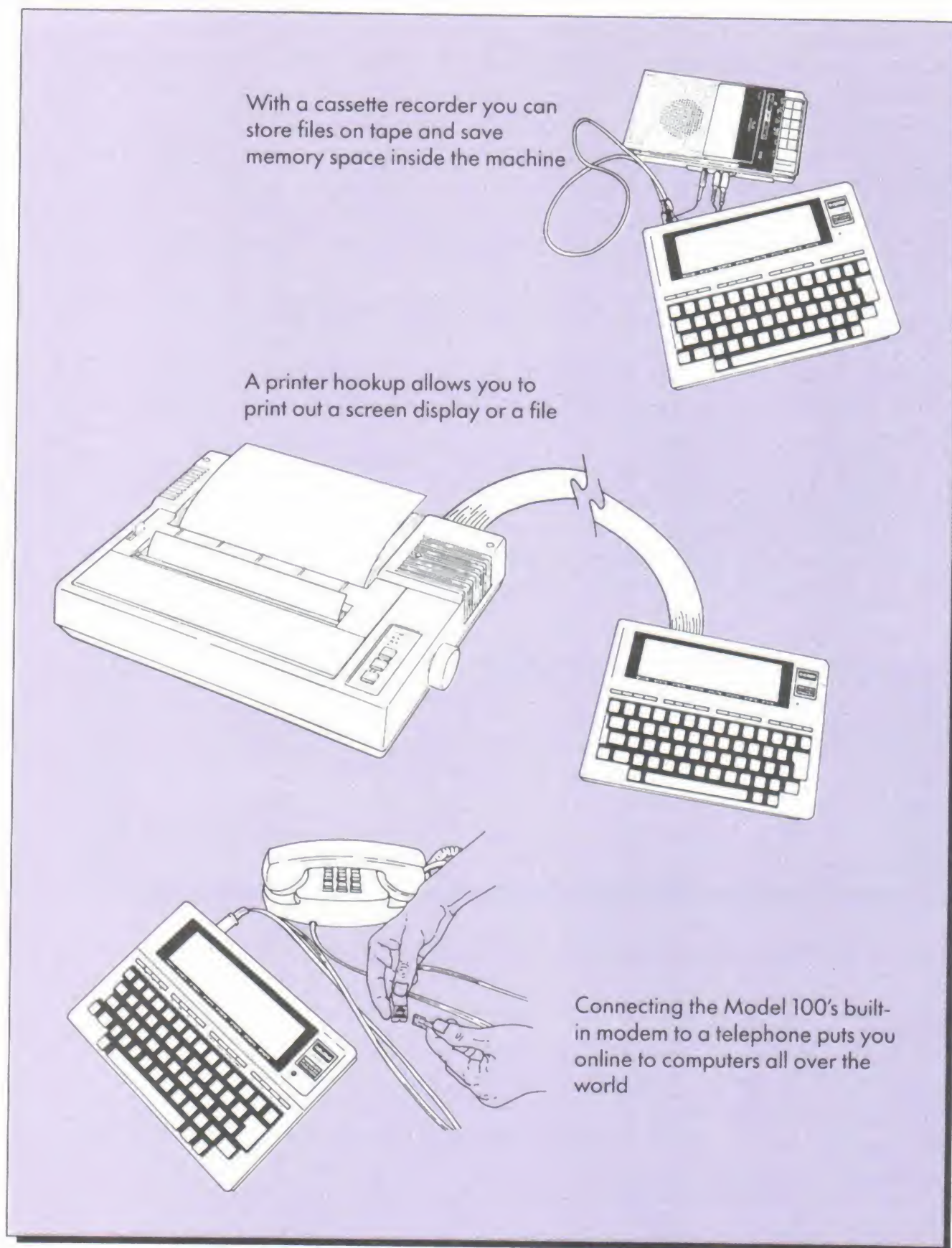


Figure 1-1. The Model 100 has several *ports*, or plug openings that allow you to connect various devices to the machine

disk in order to make room for new ones. You can also create BASIC programs that will load and run files from a cassette recorder (see Chapter 8) or from a disk (see Chapter 12). The Model 100 can also be hooked up to a printer.

The Model 100 can be hooked up to distant computers using the modem. A *modem* converts the computer's binary numbers into signals that can be transmitted over the telephone lines. The Model 100 has a built-in modem — all you need to connect it to your phone is the cable described in Chapter 9. The modem opens up the world of *telecommunications* — communications between computers using the telephone lines.

By using a short cable, you can also hook the Model 100 directly to another computer through the RS-232 port. *RS-232* was originally somebody's part number for a cable that had 25 wires connecting two 25-point pin jacks. Other manufacturers make the same cable under a different part number, and the RS-232 port is itself sometimes called a *serial* port (to distinguish it from the *parallel* port used by the printer cable).

With this brief overview of terminology behind us, we're now ready to sit down, roll up our sleeves, and learn how to put the Model 100 to work.



2

Getting Started

Concepts

- Taking a first look
- The power sources
- The keyboard
- Setting the date and time

*I*n this chapter, we begin with a “guided tour” of the Model 100: an examination of its external features, such as keys, switches, dials, ports, covers, and plugs. Then we’ll show you how to fire up your Model 100 and get it ready to go. If you have already owned your Model 100 for a while, or if your dealer or a friend has sufficiently explained its basic functions, you may want to skip this chapter and jump into Chapter 3.

Taking a First Look

Take a moment now to admire the surface features of your new equipment (see Figure 2-1). As you can see, the basic unit includes a screen display, a keyboard, and several switches and ports. The back of the Model 100 has four *ports*, or cable connectors, for attaching a cassette tape recorder, a phone line, a printer, and an RS-232C connection (to a serial printer or to another computer). (We describe these connections in detail in later chapters.) Next to the RS-232C port is a RESET button; it’s your “panic button” for use when all else fails. The RESET button cancels the current program (if one is running) and returns you to the main menu.

On the bottom of the unit are two removable panels. One houses the ROM Module Expansion Compartment, where optional cartridges can be inserted for special applications. By providing additional read only memory, this compartment allows for the possibility of adding software that uses larger amounts of memory than the Model 100 currently provides. This is

also where the optional disk drive and video interface are connected. The second panel hides the battery compartment. Just around the corner from the battery compartment is the *DC6V* (six-volt DC power) port (see Figure 2-2). If you haven't already turned on your computer, follow the instructions in the next section for loading batteries or connecting to direct current and fire her up!

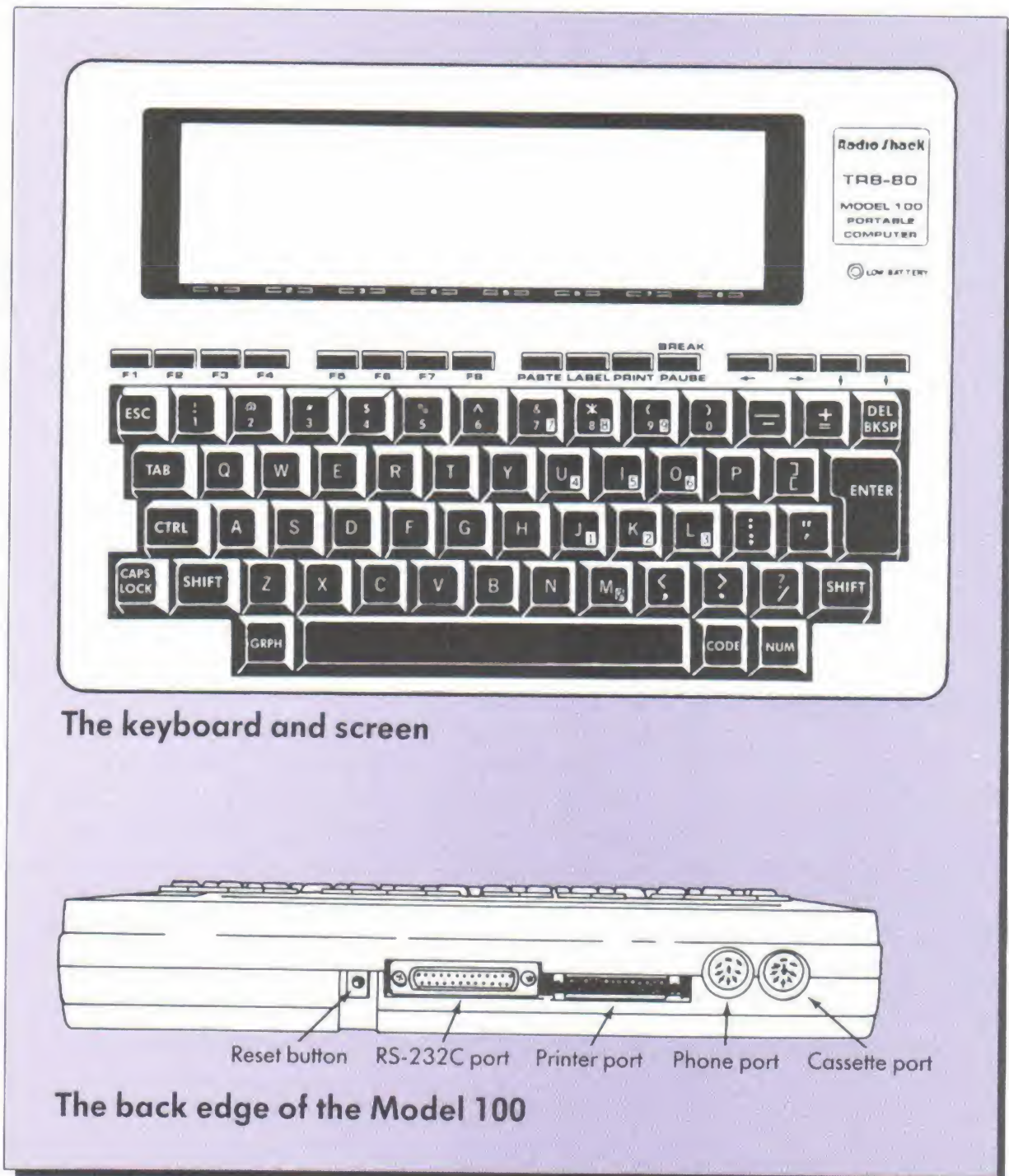


Figure 2-1. The TRS-80 Model 100 portable computer

The Power Sources

You can run your Model 100 on four size AA batteries. How long these batteries last will depend on how much you use your portable and on what you do with it. For example, it takes more battery power to telecommunicate through the modem than it does to type a letter. Model 100s with a larger amount of memory will also use more power. When the batteries are run-

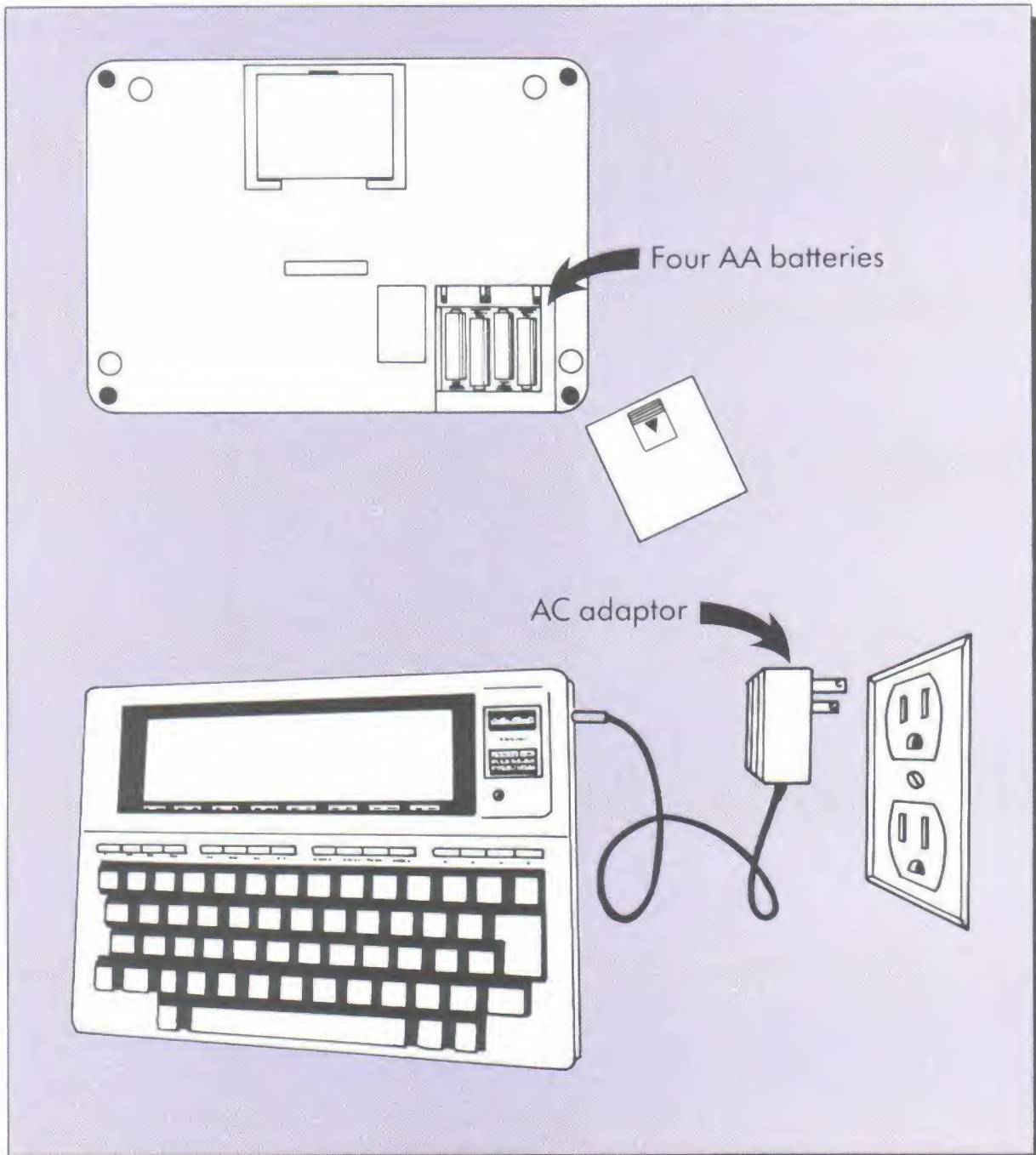


Figure 2-2. The Model 100's two sources of power

ning low, the Low Battery light in the top right corner of the keyboard will glow red. It's a good idea to keep fresh batteries in your Model 100 for instant portability and to use an AC adaptor to plug the Model 100 into any electrical outlet when working in your home or office.

You can also plug your Model 100 directly into any 6-volt direct current source, such as a boat or car battery. (Don't use a 12-volt source, though.)

Another important source of power for the Model 100 is the nicad battery installed at the factory, which lasts up to five years. This power source is used only to preserve the memory contents, not to power the screen and keyboard functions. Do not try to replace the nicad battery yourself — take the whole unit to a dealer for assistance.

You can see that the Memory Power On/Off switch is well recessed on the bottom of the unit. This switch controls the nicad battery. If you switch it off, the Model 100 will forget everything you've told it; that is, ROM will not be affected, but all user-entered files in RAM will go into computer oblivion.

If your batteries are already loaded, or you are plugged in with your AC adaptor, turn the display on now, using the On/Off switch on the right side of the Model 100.

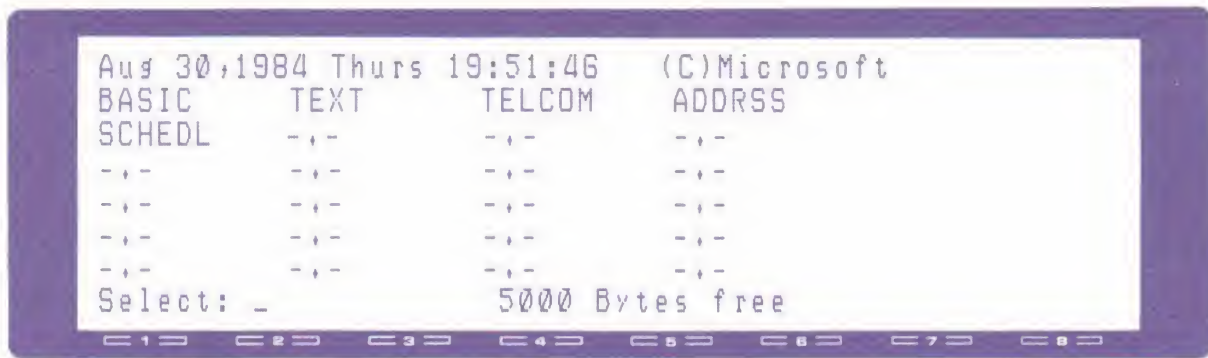
The Screen

The screen displays up to eight 40-character lines of information at a time. It consists of 15,360 *pixels* which can be individually set to display as dark or light. You can see that each character displayed on the main menu screen is actually made up of these tiny pixels. (For more detail on LCD technology, see *Hidden Powers of the TRS-80® Model 100*, by Christopher L. Morgan [New York: Plume/Waite, New American Library, 1984].)

Next to the power switch on the right side of the Model 100 is a dial labeled DISP. You can use this dial to adjust your screen display for maximum resolution.





Le Menu


The initial screen display, called the *main menu*, shows the date and time on the top line, along with Microsoft's copyright notice. (Microsoft® is the software company responsible for creating the Model 100's BASIC and other programs.) We will set the correct date and time later in this chapter. The next two lines list BASIC and the other built-in programs designed by Microsoft for Tandy: TEXT, TELCOM, ADDRSS, and SCHEDL. (Each of these programs is described in a later chapter.)

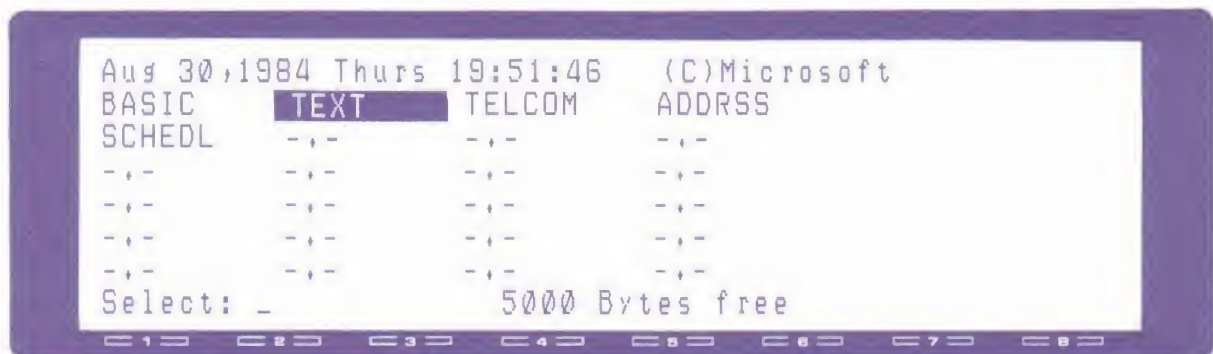


The last line of the opening display shows the prompt “Select:” and gives the number of free (unused) bytes of memory. As you add files to memory, the number of free bytes will drop. Keep your eye on this figure. You will periodically need to erase old files to make space for new ones. (In later chapters, we’ll tell you how to copy files from the memory onto cassette tapes, to other computers, or to the optional disk drive: with these options, you can easily save your files so that you can leave room in RAM for new additions.)

The Keyboard

The keyboard has all the keys of a conventional typewriter, plus some special function keys. Try pressing the arrow keys (, , , and ) and watch the shaded cursor jump to different filenames on the menu.

In order to test a few more special keys, we’ll start the *TEXT* program by positioning the cursor over the word *TEXT* on the menu and then pressing .



The screen display changes to show the prompt “File to edit?”. Here we need to enter a filename that is no longer than six letters such as FUNKEY. The word *enter* as used in this book means to type something and then press

the **ENTER** key, the large key to the right of the alphabet keys. We want to *enter* FUNKEY, so we type in FUNKEY and then press **ENTER**.

```
File to edit? FUNKEY
```

Again the screen display changes to show the shaded cursor blinking in the top left corner of the screen. The cursor shows you where the next character you type will appear. Try typing a word or phrase now.

```
Twas brillig, and the slithy toves . . .
```

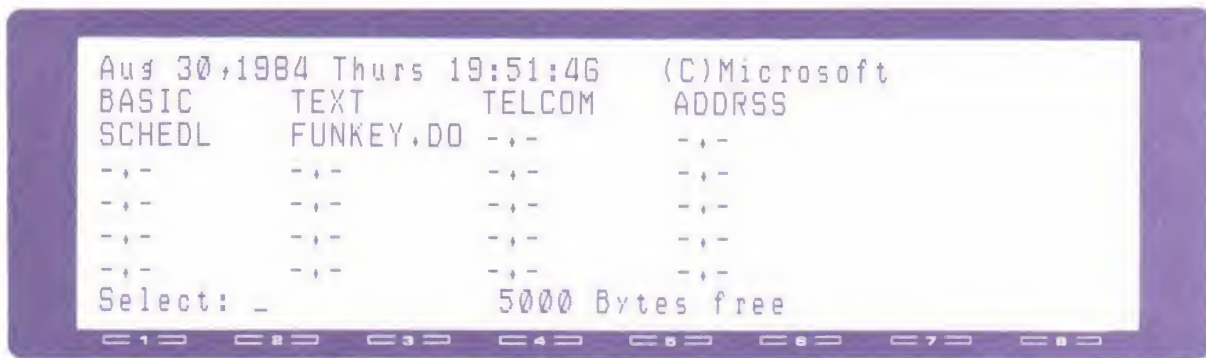
Notice that, as you type, you needn't press **ENTER** at the end of each line on the screen; words automatically *wrap around* to the next line when the text reaches the right edge of the screen.

Press **ENTER** to start a new paragraph, then try typing a few characters while holding down the **GRPH** key. Thirty-two special characters are available through this graphics function. (Unfortunately, most printers cannot print these special characters; they are useful only in screen displays.) Similarly, thirty-two additional characters are available by holding down the **CODE** key while striking other keys. Try some now.

The top row of keys on the Model 100 shows more built-in special functions. Press **LABEL** to view a list of the functions assigned to the keys **F1** through **F8**. This is how they should look in the TEXT program:



Each of these keys has a different definition, depending on the program: they're different in BASIC and SCHEDL, for example. (All the functions for different programs will be explained in the appropriate chapters of this book.) For now, exit the TEXT program and, simultaneously, save the file to memory by pressing function key **F8**. The menu returns to the screen, showing the new filename FUNKEY.DO in addition to the five built-in programs.



The Model 100 automatically adds the suffix “.DO” to any *text filename*; the suffix “.BA” is added to any *BASIC* program file you create.

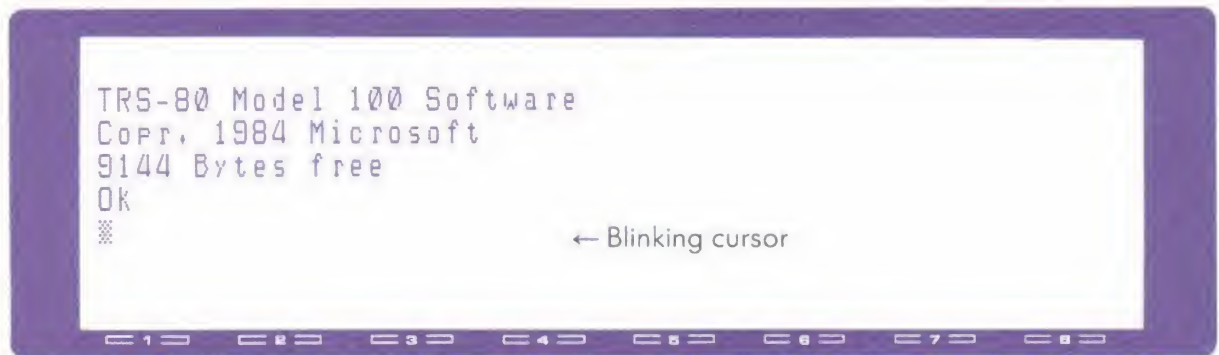
Setting the Date and Time

The Model 100 has many small features that make it easier and more convenient to use. One of these is the built-in clock. You can use the clock in your programs to automatically print the date on a document, for example. Your Model 100 can even act as a travel alarm clock!

To set the date and time, we will use the BASIC program. BASIC is mostly used for programming, but it’s also used to send simple messages to the Model 100. And the nice thing is you don’t have to be a programmer to use BASIC (see Chapter 6). Let’s try it now. First, position the cursor over the word *BASIC* on the main menu and press **ENTER**.



Now the screen blinks and displays four lines that indicate you are now in BASIC:



(The number of bytes free will vary depending on how much memory capacity your Model 100 has and how much of it is used by any other files you have already created.)

The blinking cursor appears below the word *Ok* on the screen. This is where your next entry will be displayed as you type. To set the date, simply type the following:

```
DATE$="09/01/84"
```

Be sure that there are no extra spaces between words in your entry. Always set the date in quotation marks; use two digits for month, day, and year, separated by slashes; and press **ENTER** after typing all the characters. When you press **ENTER**, BASIC reads what you typed, sets the date, and responds "Ok". To check that the date has been set correctly, type:

```
PRINT DATE$
```

(Use a space between the words this time.) BASIC reads what you typed and responds by displaying the date. If the date displayed is not the date you expected, try entering the date again, being very careful about the way you type it in.

To set the time, type `TIME$=`, followed by the hour, minute, and second, separated by *colons*. End with a closing quotation mark and press **ENTER**:

```
TIME$="23:45:01"
```

To check the time now, type:

```
PRINT TIME$
```


If the time displayed is incorrect, try entering `TIME$ =` again.

Once you have set the date and time, press **F8** to exit from BASIC and return to the main menu, where the current date and time should now be displayed on the top line.



The Model 100's built-in clock and calendar, once set to the correct date and time, will continue to operate reliably as long as the internal nicad battery operates.

A Note About File Size

The bottom line of the main menu shows how many bytes of memory are free. This number will decrease every time you create and save a new file. As we noted earlier, it's a good idea to watch this "bottom line" when you create new files. Eventually, your memory space will become filled and you will need to delete some of the old files to make room for new ones.

Before you delete useful files, you can copy them onto cassette tapes or transmit them to another computer. Later, when you need them, you can copy them back into the Model 100 memory. (See Chapters 7 and 11.)

If you run out of memory space, the Model 100 will display the error message OM.

Moving On

You've learned how to power up your Model 100 and set the time and date. Now we're ready for a more ambitious project: learning to use the built-in text-editing program, TEXT. In many ways, this program is the key to all the Model 100's functions; it is also the program you will probably use the most often.

M DEL 100



3

A Terrific *TEXT* Editor

Concepts

- Starting a text file
- Naming a file
- Typing text
- Cursor movement
- Correcting errors
- Saving the file
- SELECT and PASTE a block of text
- PASTE text to a second file
- FINDing text

Writing tools have come a long way since the days of stones and chisels, but one aspect of writing has been a problem for every writer since the beginning: correcting errors. Some of us still remember the days of ink eradicators and hard erasers, and we've all had to retype the same letter more than twice before it was free of typos. Copying machines ushered in a major revolution, allowing us to make a clean duplicate of an original full of white correction fluid; but even so, producing error-free documents can be time-consuming.

With the introduction of electronic word processing, we simultaneously acquire two luxuries: we can correct errors and make other changes to parts of the text without retyping the whole page, and we can print out any number of original copies!

One of the five built-in programs that come with the Model 100 is called *TEXT*. This is the text editing program that makes it easy to type, edit, and save any kind of text file. We can type ordinary paragraphs of text, such as memos and letters. We can also type text files that have a special format, such as a list of names and addresses, and we can even use *TEXT* to help us type BASIC programs and data files. (See Chapter 6 for more on this.)

Once we have typed and saved our text, we can view it and edit it whenever we like. We can also print it out or copy it onto a cassette tape. We

can even “telecommunicate” it to another computer through the telephone lines or through the RS-232 interface.

In this chapter, we show you how to create a new text file, how to edit it, save it, review it, and erase it entirely. In other chapters, we will show you how to send your text to a printer, a cassette recorder, or another computer.

A Quick Memo

Here we want to introduce you to text editing. The methods of entering and editing text described here apply to all types of text files and to BASIC program files when you are in the EDIT Mode.

We’ll assume that you have never used a computer before, so we’re going to take you step by step through the process of creating a new file, saving it, getting into it again to change it, and, finally, deleting the file from memory.


We will begin by typing a memo — the simplest, shortest document we could think of. Our text file looks like this:

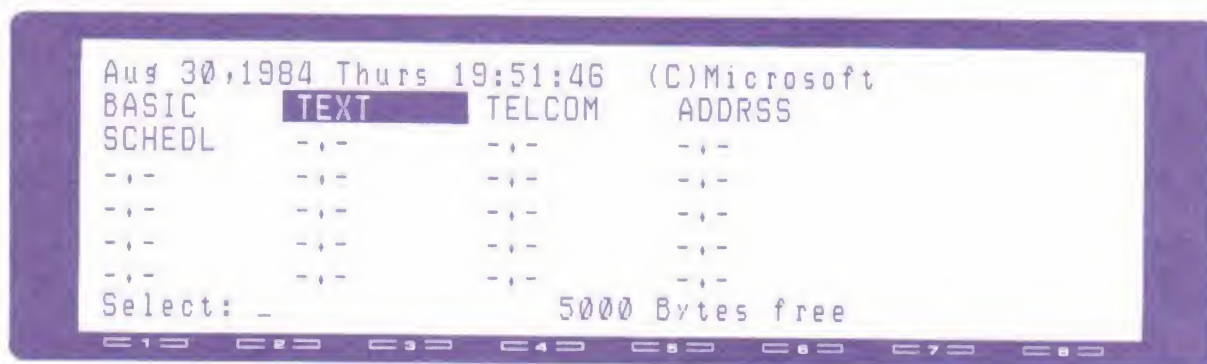
TO: R. Thompson◀

◀
Regarding that new ergonomically
designed chair you requested -- GREAT
IDEA! Such a chair could substantially
improve your throughput. Thanks for the
tip - I’ll order one for all department
heads.◀

◀
M. Cobb◀

Starting a Text File

The first thing you need to do to create a memo is to enter the TEXT program. Turn on your Model 100 and press the  to position the cursor over the word *TEXT* on the main menu.



With the cursor over the word *TEXT*, press **ENTER**. The main menu disappears and you see the prompt “File to edit?”. It’s time to pick a name for this text file.

Naming the File

Two rules apply in naming files: (1) the name may be no longer than six letters, and (2) it must begin with a letter of the alphabet. Filenames must not *begin* with a number or symbol, though numbers and some symbols may be included after the first letter of the name. Additionally, two different files cannot have the same name.

LETTER, CSHFLW, A12345, M9/2, and L34-50 are all examples of acceptable filenames. When you finish typing and you save your file, the Model 100 will list the file on the main menu, adding the suffix “.DO” to show that this is a text, or DOcument, file.

In this example, we chose to name the file MEMO, so, after the prompt, we type in the name and press **ENTER**:

File to edit? MEMO

Again the screen clears and a flashing € symbol appears in the top left corner. The € symbol is not the cursor; rather, it shows you where the end of the file is. Of course, because this is a new file with nothing in it, the end of the file is at the beginning of the file. (If you had entered the name of an existing file, then the contents of that file would be displayed on the screen, and the € would appear at the end of the text.)

Typing Text

Let’s take the plunge and start typing. Begin by typing in the first line of the memo and pressing the **ENTER** key. For uppercase letters, simply hold down the **SHIFT** key and press the letter. Pressing **ENTER** after typing the first line causes a carriage return, which brings you to the second line (actually the second *paragraph*) on the screen. Press **ENTER** again to skip a line before entering the text of the main paragraph.



In order to edit this memo conveniently and print it at any width you choose, type each entire paragraph without pressing **ENTER**. Press **ENTER** *twice* at the *end* of a paragraph in order to end the paragraph and skip one line before the next paragraph. (If you have never used a word-processing program before, this may be difficult to get used to, because it's different from a typewriter.)

TO: R. Thompson

Regarding that new ergonomically designed chair you requested -- GREAT IDEA! Such a chair could substantially improve your throughput. Thanks for the tip - I'll order one for all department heads.

M. Cobb

← Don't press **ENTER**
 ← Just keep typing until end of paragraph

← Press **ENTER** here
 ← Press again to skip line

That's all there is to it! Now you should review the text to see if everything is just the way you want it.

However, you can't see the beginning of the file because it has *scrolled* off the top of the screen. To move backwards through the file so you can return to the first line, press the **↑** key in the upper right corner of the keyboard (see Figure 3-1).

You can see that each time you press **↑**, the blinking cursor moves up one line on the screen. If you hold the **↑** key down, the cursor will "run"

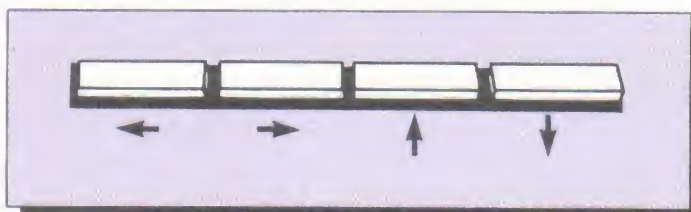


Figure 3-1. The arrow keys move the cursor

upward on the screen. Lines that previously scrolled off the top will scroll back onto the screen until you have reached the beginning of the file.

You can also move the cursor faster, making it jump seven lines at a time. To do this, hold down the **(SHIFT)** key while pressing **(↑)** or **(↓)**.

You can also move the cursor directly to the beginning or end of the file by using the **(CTRL)** key. The **(CTRL)** key operates like the **(SHIFT)** key — you must hold it down while you press the next key. To jump back to the beginning of the file, hold the **(CTRL)** key down as you press the upward-pointing arrow key. You can see that pressing **(CTRL)(↑)** does not print any characters on the screen; it simply jumps the cursor to the beginning of the file.

You have just seen how **(↑)** and **(↓)** are used to move the cursor up and down lines in the file. These keys move the cursor only and do not change the text. Similarly, **(→)** and **(←)** move the cursor to the right and left along a line. The cursor will “wrap around” to the next line up or down when you reach the end (or beginning) of a line.

Cursor Movement and the **(CTRL)** Key

As an alternative to the arrow keys, the **(CTRL)** key can be used to move the cursor. The box below lists all the ways to move around on the screen when in a text file.

Cursor Movement in TEXT

Keystrokes	Cursor Movement
(↑) or (CTRL)E	Up one line
(↓) or (CTRL)X	Down one line
(→) or (CTRL)D	Right one letter
(←) or (CTRL)S	Left one letter
(SHIFT)(↑) or (CTRL)T	Up one screen
(SHIFT)(↓) or (CTRL)B	Down one screen
(SHIFT)(→) or (CTRL)F	Right one word
(SHIFT)(←) or (CTRL)A	Left one word
(CTRL)(↑) or (CTRL)W	To beginning of file
(CTRL)(↓) or (CTRL)Z	To end of file
(CTRL)(←) or (CTRL)Q	To beginning of line
(CTRL)(→) or (CTRL)R	To end of line

Why use **CTRL** when **↑** does the same thing? This may seem confusing to some of you, but old pros will recognize these commands from a popular word-processing program for microcomputers, WordStar®. Back in the ancient days of microcomputing, many keyboards had no arrow keys, and the **CTRL** key commands were used instead.

Why do we use E, X, S, and D instead of U, D, L, and R (for “Up”, “Down”, “Left”, and “Right”)? The E, X, S, and D keys were selected for their *relative positions on the keyboard* (see Figure 3-2). Notice that the E key is *above* the X key, and the S key is to the *left* of the D key. Similarly, the key to the left of S will jump one whole *word* left at a time, and the key to the right of D will jump one word right.

If you aren’t already experienced in WordStar, there’s probably no point in learning these **CTRL** E-style commands. The arrow keys are easier to remember, so you can stick with them.

Correcting Errors

Try all these cursor movements. You can see that they do not affect your text at all; they merely move the cursor to various places in the text. By moving the cursor, you can review your entire file and make corrections or additions to your text.

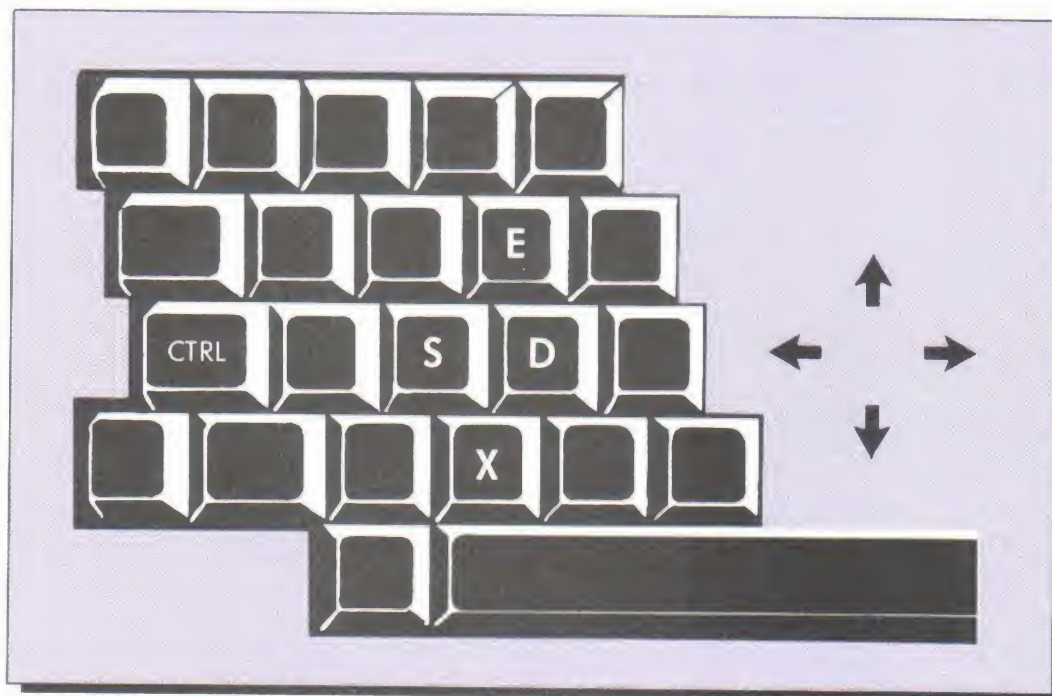


Figure 3-2. An alternative to the arrow keys: the ESXD diamond

By now you may have noticed that your file has one or two typos that you need to correct. Or maybe you're a perfect typist but want to change something you've entered.

Let's change the signature on our memo from M. Cobb to Michael Cobb. First, position the cursor at the period after the initial M, then type the letters *ichael*:

```
TO: R. Thompson
```

```
Regarding that new ergonomically  
designed chair you requested -- GREAT  
IDEA! Such a chair could substantially  
improve your throughput. Thanks for the  
tip - I'll order one for all department  
heads.
```

```
Michael. Cobb
```

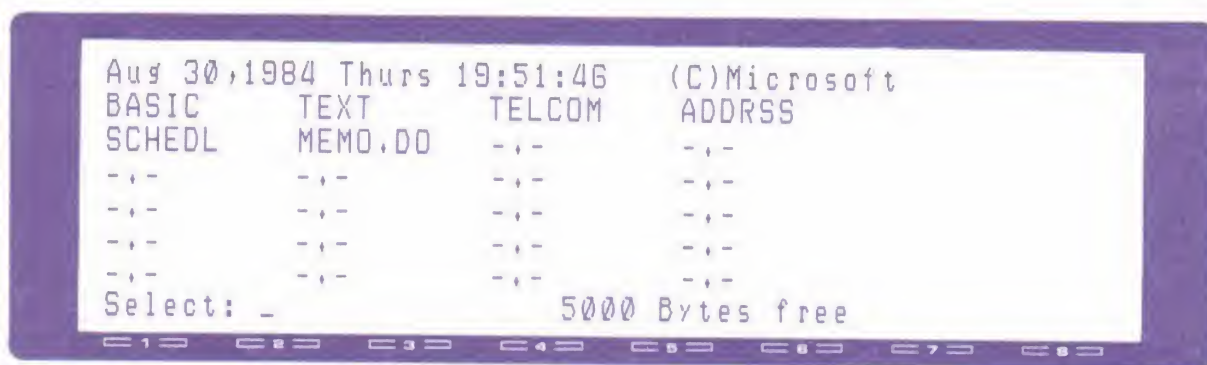
In other words, to *insert* characters, simply move the cursor to the exact position where you want the new characters to appear and start typing. Any typed character will appear on the screen at the cursor position, and all following text will shift to the right as you type. You can insert letters, words, sentences, or paragraphs. To insert a blank line, position the cursor at the end of the last letter on the line above and press **ENTER**; a carriage return will be inserted. To add text at the *end* of a file, press **CTRL** **↓** to move to the end of the file and start typing.

Now we want to delete the period that still appears after the word *Michael*. When editing typed text, you can position the cursor over the letter you wish to delete and press **SHIFT** **DEL/BKSP** (delete/backspace). Holding down the **SHIFT** key as you press **DEL/BKSP** deletes the character *at* the cursor location. To delete the character to the *left* of the cursor, press the **DEL/BKSP** key alone. (You can do the same thing with **CTRL** **H**.) This is most handy when you are in the process of typing new text and notice that you have just typed the wrong letter.

Deleting one character at a time is simple enough, but it can get tedious when you need to delete whole lines or paragraphs. We'll show you how to handle blocks of text in the next section.

Saving the File

To save the text file you have typed, simply press **F8**. The main menu will appear on the screen, but now MEMO is in memory and is listed on the menu.

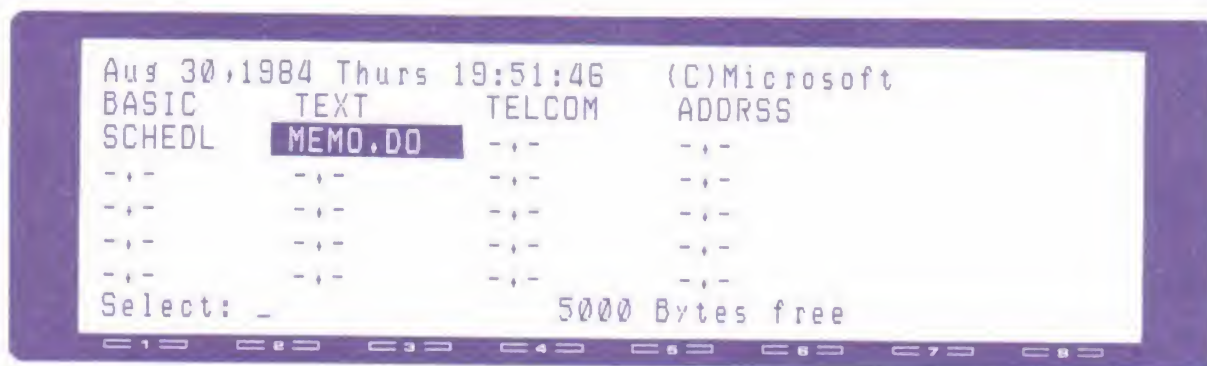


Working With SELECTed Text

In this exercise, we will revise the MEMO file we created previously and move part of the text to a new file called REPLY. To do this, we must SELECT and PASTE blocks of text.

A block of text can be a word, a line, several lines, or a whole file. You can select and delete the block with one command, or you can move or duplicate it to another location in the file, or to an entirely new file.

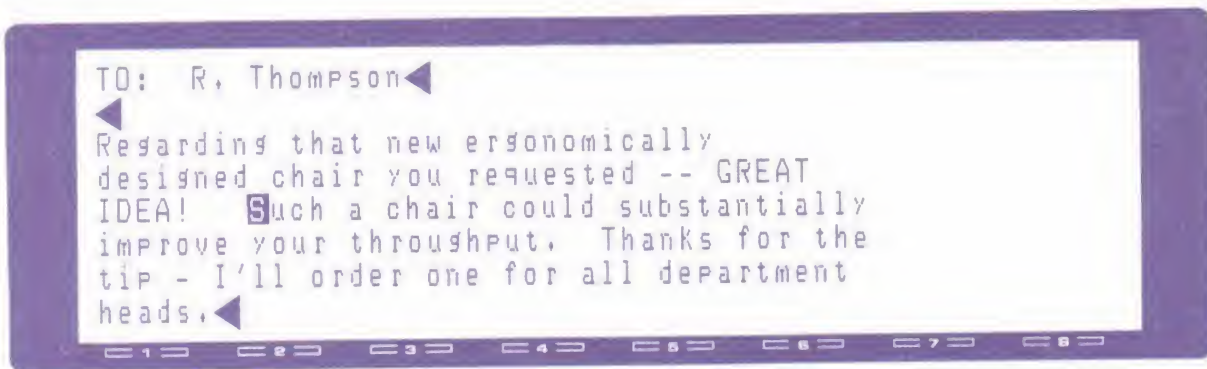
Begin by returning to the file MEMO.DO, which you saved to memory in the last exercise. Position the cursor over the filename on the main menu and press **(ENTER)**:



Deleting a Block of Text

In copying, moving or deleting a block of text, we begin by using the SELECT key to identify the block. SELECT is one of the function keys. (Remember, to see what each of the function keys will do in the TEXT program, press the **(LABEL)** key.)

First, position the cursor over the S of "Such a":



Now press (F7) SELECT. This marks the beginning of the block you want to operate on. Now use the (→) key to move the cursor to the right along the line. As the cursor moves, the text it passes over will appear in *reverse video*. This is the *marked block* of text. If you use the (↓) key to move down a line, you'll see that the entire line is "marked" or changed to reverse video at once. Conversely, if you go left with (←) or up with (↑) into the marked text, you'll find that it becomes "unmarked", or restored to normal video. By using the cursor keys, you can thus adjust the marked block to exactly the size you want.

Position the cursor before the word *Thanks* so that the period after the word *throughput* is included.

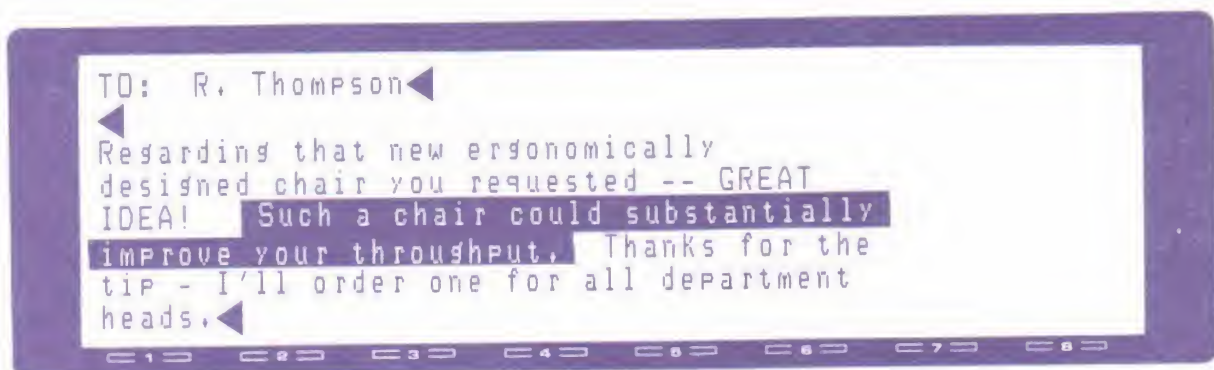
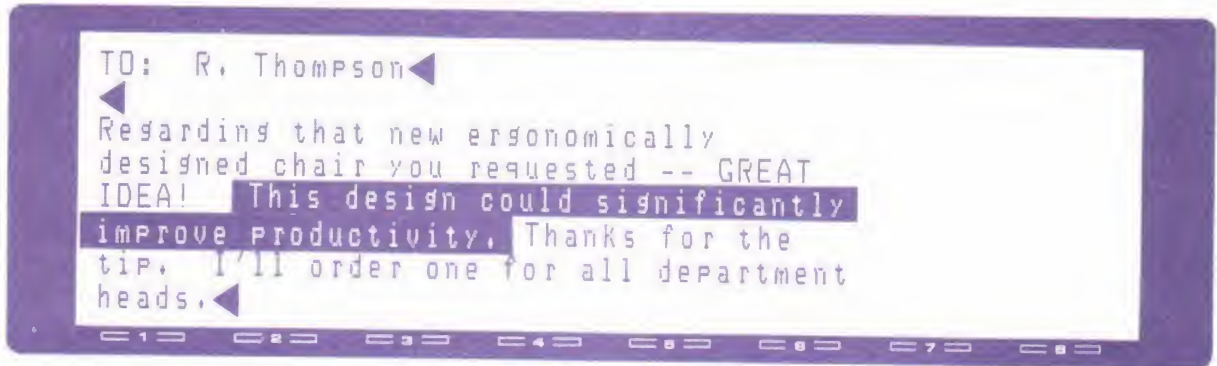


Figure 3-3. The paste, label, print, and pause keys

Now we want the Model 100 to “cut”, or delete, this text out of our file. To do this use **(F6)**, the CUT function key. The marked text disappears! It is not gone completely, however. It is in a special holding place, or “paste pocket”, in memory.

With the cursor still positioned before the word *Thanks*, type “This design could significantly improve productivity.”



(PASTE) Text to a Second File

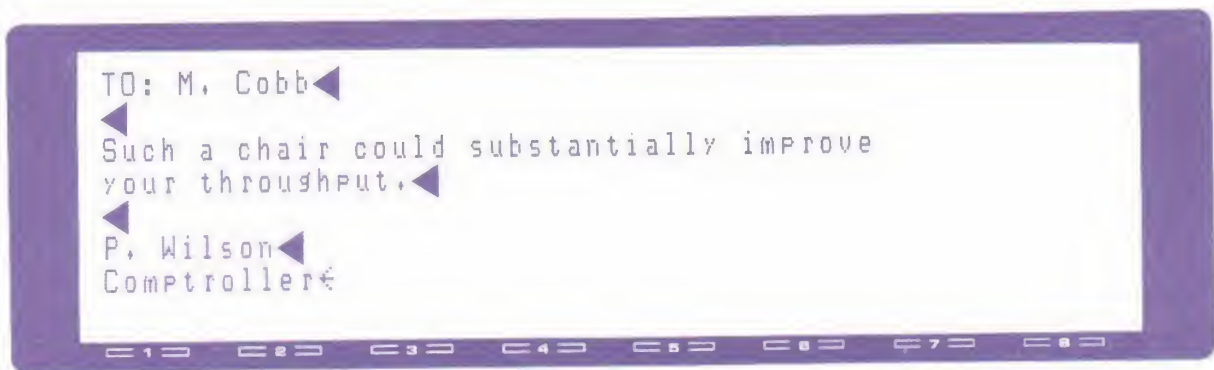
Now, exit this file by pressing **(F8)**; then, open a new file called REPLY. (Follow the instructions given at the beginning of this section.) Begin by typing:

```
TO: M. Cobb
```

To position the cursor at the point where the text will begin, press **(ENTER)** twice. Then press the **(PASTE)** key and watch the contents of the Model 100 “paste pocket” appear on the screen:

```
TO: M. Cobb
Such a chair could substantially improve
your throughput.
```

You can finish this exercise by typing in a signature.



Now, you can save this file to memory, using the same steps as before.

Here is a summary of the steps for SELECTing text and copying or deleting that text.

Selecting Text

Step 1 Position the cursor at the beginning of the block. Enter the SELECT mode by pressing (F7).

Step 2 Move the cursor to the end of the block using the arrow keys. To copy the text into the Model 100 “paste pocket”, press (F5) or (F6). (Use (F5) if you wish to duplicate, or *copy* the text elsewhere, and use (F6) to *move* a block to a new location. Remember, when you *move* a block, it disappears from its original location.)

Step 3 Move the cursor to the position where you want the block to appear and press the (PASTE) key.

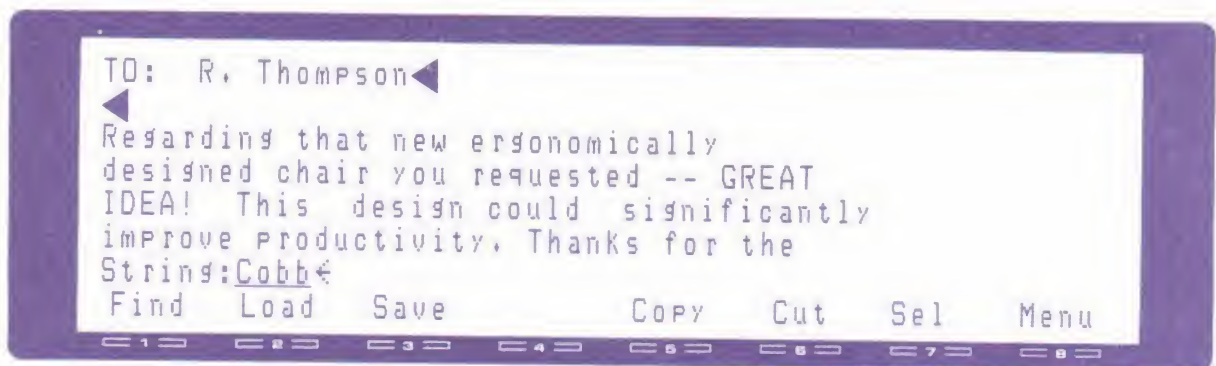
When SELECTing text, you can use (SHIFT) or (CTRL) with (→) or (↓) to jump the cursor ahead. You can also work in reverse, that is, position the cursor at the end of the block and press (F7), then use (←) or (↑) to move backward to the beginning of the block and press (F5) or (F6).

Whatever you put in the paste pocket will remain there until you replace it by making a new block selection. You can turn the Model 100 off, and the paste pocket will still be there when you return. You can also exit the current file, enter another file, and press (PASTE) — whatever you last placed in the pocket will be copied onto the screen at the cursor position. (You can even PASTE into BASIC programs in EDIT Mode; see Chapter 6.)

Using the FIND Function

The Model 100 offers text-editing capabilities that make it suitable for all kinds of documents, large or small. One particularly useful feature is the FIND function, which allows you to “search” for a particular string of text. This can be particularly helpful when dealing with longer files, because you can search for text anywhere in the file; the Model 100 will then jump there much more quickly than with the standard cursor movements.

Let’s go back to our MEMO file for a moment. (Or feel free to use your own text file; perhaps you have created some file on your own by now.) Let’s search for the word *Cobb*. Press **(LABEL)** to display the function key definitions in TEXT Mode. Now press **(F1)** for FIND. At the bottom of the screen, you will see the prompt “String:” displayed. The prompt means literally “What is the string of characters you wish to find?” Simply type in the text you wish to FIND and press **(ENTER)**:



The Model 100 will quickly search through the file, and the cursor will jump to the first letter of the string you entered:



You can enter up to twenty-four characters in the FIND string. The characters can be letters, numbers, or symbols.

Global Search and Replace

If you've had experience with word processors before, you probably turned to this section expecting a powerful one-word editing command. Sorry, there is no single "find and replace" command in the Model 100 (unless you buy an additional text-editing package). The best suggestion we can offer is this procedure:

Find and Replace

- Step 1 Move to the beginning of the file by pressing **CTRL** **↑**.
- Step 2 To find the first occurrence of the word or phrase you wish to replace, press **F1**. In response to the prompt, "String:" type the word or phrase and press **ENTER**.
- Step 3 When the cursor jumps to the beginning of the found word, delete the unwanted characters using the **DEL/BKSP** key. Then, type in the new word or phrase.
- Step 4 With the cursor at the end of the newly inserted phrase, enter SELECT by pressing **F7**. SELECT backwards to the beginning of the new word or phrase you just typed in using **←**.
- Step 5 Copy the new phrase into the "paste pocket" by pressing **F5**.
- Step 6 Find the next occurrence of the old phrase by pressing **F1** **ENTER**.
- Step 7 Delete the phrase using **DEL/BKSP**, and insert the new phrase by pressing the **PASTE** key. Repeat steps 6-7 to the end of the file.

The best way to learn anything is to practice. Try typing in your own document to familiarize yourself with the TEXT program. In later chapters, we'll show you how to send your TEXT files to another microcomputer, or to a cassette tape for storage, as well as how to get a printout of your text files.



4

Getting Organized

Concepts

How SCHEDL works with NOTE.DO
Creating NOTE.DO
Using SCHEDL

*T*ime management is one of the most difficult tasks to master. Businesses spend millions of dollars annually for time-management specialists to teach their employees how to effectively allocate and organize their time. On a more individual level, many books have been written to show us how to more effectively structure time in our daily lives. Even if we perform roughly the same activities every day, chances are we keep a special calendar *somewhere* to remind us of birthdays, holidays, and other special events. But many of us write our schedules on little slips of paper, which are easily lost and hard to organize effectively. The Model 100 offers a simple but remarkably powerful solution to the problems of time management with its built-in SCHEDL program. Let's find out how to use the Model 100 to get organized.

The Model 100's built-in SCHEDL program works with a special file that contains your list of things to do: NOTE.DO. NOTE.DO is a file you create in the TEXT program. After going through the basic steps of building a NOTE.DO file and using SCHEDL, we'll show you examples of NOTE.DO files that other executives have used.

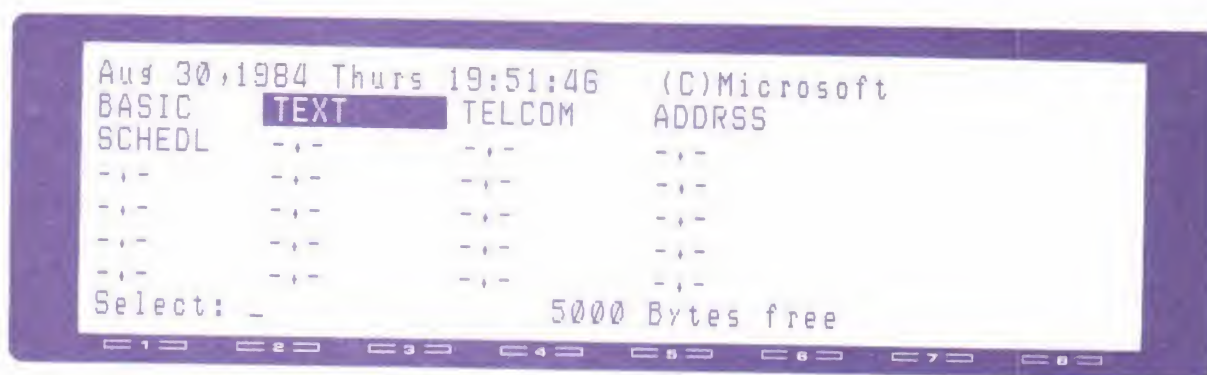
Creating NOTE.DO

In this chapter, we will create a NOTE.DO file that looks like this:

```
NOTE.DO          30 AUGUST 1984
DATE  TIME      DESCRIPTION
-----
09/01 08:00A Meet Bob
09/01 09:00A Staff Meeting
09/01 12:00P Lunch with rwl
09/01 02:00P First interview
09/01 04:00P Last interview
09/02 08:00A Dentist
09/02 10:00A Board Meeting
09/02 01:00P Lunch w/ ACM
```

With this information as our base, we will use the SCHEDL program to find out what we have to do tomorrow. If you want, you can type your own schedule into your NOTE.DO file and follow our steps, using your own search words.

To create the NOTE.DO file, first position the cursor over the word *TEXT* on the main menu and then press **ENTER**.



When the Model 100 asks you for a filename, type NOTE.DO and press **ENTER**:

```
Name of file to edit?  NOTE.DO
```

Now the screen clears and the cursor appears in the top left corner, ready for your entries. (If NOTE.DO already exists, the screen displays the first eight lines of the file.)

Let's begin by typing the filename and date on the top line:

```
NOTE.DO          30 AUGUST 1984
```


Whenever we update a file, it is a good practice to enter the current date on the top line. In this way, whenever we open a file, we can tell at a glance how current it is. (This is just one way of designing a NOTE.DO file; other formats are shown later in this chapter. Of course, your file needn't look like anyone else's.)

Next, we can begin typing notes about the things we plan to do. In our case, let's enter three types of information for each item in our list — the date, the time, and task description. We type these three headings on the second line of the screen, using the **(TAB)** key between each:

```
NOTE.DO      30 AUGUST 1984
DATE        TIME    DESCRIPTION
```

If you like, type a row of hyphens on the third line to separate the headings from the entries that will follow. This will take up 40 bytes of memory space, but it will make this list easier to read when it is printed out. If you are more concerned with conserving memory space, you may wish to begin with the actual schedule entries, such as the following.

```
NOTE.DO      30 AUGUST 1984
DATE        TIME    DESCRIPTION
-----
09/01      08:00A    Meet Bob
```

By using the **(TAB)** key to separate columns, we create a neat, easy-to-read list and at the same time save memory space. How? The **(TAB)** key uses only *one* byte of memory, although more than one space may appear between the words on the screen.

We are entering the date as MM/DD (month and day). You needn't be as formal as this. You could enter the date as "09/01" or "9/1" or "1 Sept" or "9-1-84", and so on. The important thing is to choose one method and stick with it so that when you use the FIND or LFND commands, you will locate all the appropriate lines in one search.

The time is entered as HH:MM (hour and minute), followed by an *A* to indicate "AM". Decide for yourself whether you want to use a 24-hour clock or type *A* or *P* after the time entry. As with the date format, you can enter the time in any way you wish.

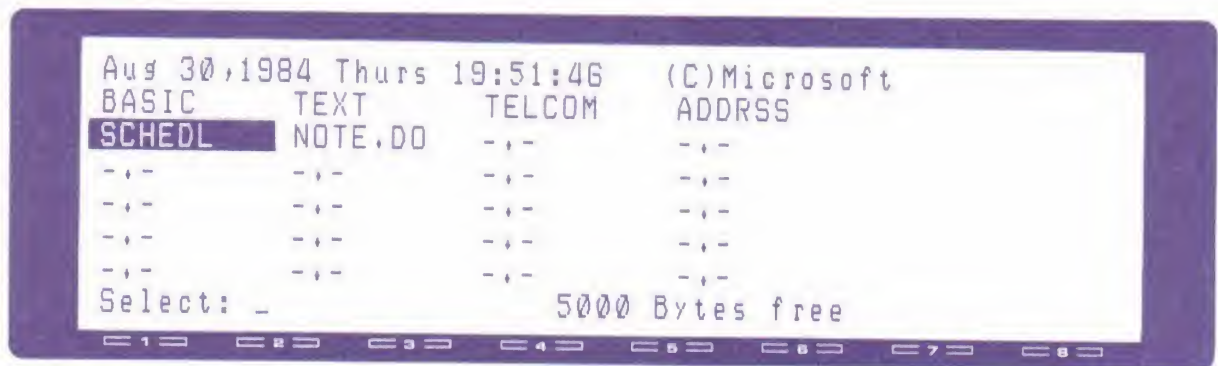
Make a few more entries of your own, using your own format. A full week of entries might look like this:

```
NOTE.DO          30 AUGUST 1984
DATE    TIME    DESCRIPTION
-----
09/01    08:00A  Meet Bob
09/01    09:00A  Staff Meeting - Prepare
monthly sales report
09/01    12:00P  Lunch with rwl
09/01    02:00P  First interview
09/01    04:00P  Last interview
09/02    08:00A  Dentist
09/02    10:00A  Board Meeting - Prepare
equipment purchase proposal
09/02    01:00P  Lunch w/ ACM
```

Notice that two items on the list, a staff meeting and a board meeting, run onto a second line. In typing this line, we did not press **ENTER** until after typing the entire entry. In this way, we can see both lines of information whenever any word in the lines is “found” by the SCHEDL program, as you will see below.

Using SCHEDL

Once you have made some entries in the file NOTE.DO, you can use the SCHEDL program to select specific lines in the file and either display them on the screen or print them out. First, exit to the main menu by pressing **F8**. To use the SCHEDL program, begin by positioning the cursor over the word *SCHEDL* on the main menu and pressing **ENTER**.



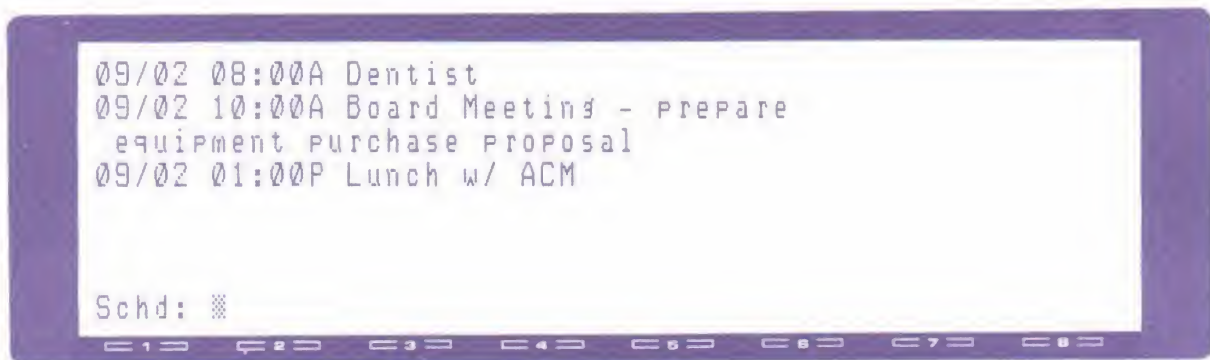
The screen displays the prompt “Schd:” on the top line, and the bottom line shows the three possible actions you can take with SCHEDL: Find, Lfnd, or Menu.



Let's try the first option listed (Find) by pressing **(F1)**. The top line now displays the word *Find*. (We could have typed the word *F-I-N-D*, but pressing **(F1)** is a lot faster.) For our first example, let's list only those lines that include the date 09/02. To do this, press **(F1)**, type the date 09/02, and press **(ENTER)**:

Find 09/02

The screen displays all the lines from NOTE.DO that include the date 09/02.



Notice that you would *not* have found these lines if you searched for "9/2" but you *would* find them if you searched for "9/02". You can see why you need to be consistent in making your entries in NOTE.DO. If the date is important to you, be sure there is a date on every line. If dollar amounts are important, then be sure they all are entered with some character you can search for — a dollar sign or a decimal point.

Notice that the second item on the list, a board meeting, runs onto a second line. In typing this line, we did not press **(ENTER)** until after typing the final word, *report*. In this way, we can see both lines of information, even though the search word "09/02" does *not* appear on the second line. If we had typed **(ENTER)** after the word *prepare*, the second line of information would *not* be listed here.)

You can use the SCHEDL program to list the entire contents of NOTE.DO by pressing **(F1)** and then pressing **(ENTER)**. Because you do not specify *what* to find here, the SCHEDL program finds *every* line!

Schd: Find

First, SCHEDL displays the first six lines from NOTE.DO, and the bottom line shows that you can press **(F3)** for More or **(F4)** to Quit, that is, quit the “Find” operation.

```
NOTE.DO          30 AUGUST 1984
DATE  TIME      DESCRIPTION
-----
09/01 08:00A Meet Bob
09/01 09:00A Staff Meeting - Prepare
monthly sales report
09/01 12:00P Lunch with rwl
                        More  Quit
<1> <2> <3> <4> <5> <6> <7> <8>
```

Press **(F3)** (or *any* key) to display six more lines. When you reach the end of the list, SCHEDL will again display the prompt “Schd:” at the bottom of the screen.

```
09/01 02:00P First interview
09/01 04:00P Last interview
09/02 08:00A Dentist
09/02 10:00A Board Meeting - Prepare
equipment purchase proposal
09/02 01:00P Lunch w/ ACM
Schd: ✖
Find                                Lfnd                                Menu
<1> <2> <3> <4> <5> <6> <7> <8>
```

There are only two other options available under SCHEDL: Lfnd and Menu. LFND operates exactly the same way as FIND, except that it lists the found lines on the printer instead of on the screen. (If you have a printer, see Chapter 7 for more information about the LFND function.) Finally, when you are finished using SCHEDL, press **(F8)** “Menu” to return to the main menu.

Managing NOTE.DO

Maintaining and managing the NOTE.DO file may require some thought. Note that you can only have *one* file named NOTE.DO and that only NOTE.DO can be used with the SCHEDL program.

To update the NOTE.DO file, simply enter NOTE.DO from the main menu and edit it as you would any text file. You cannot edit NOTE.DO through the SCHEDL program.

If your NOTE.DO file becomes very large, you might want to divide it into three files: NOTE.DO, FUTURE.DO, and DONE.DO. Using this approach, old lines in the file NOTE.DO are copied into the file DONE.DO using the Model 100 CUT and PASTE operations. New lines are pulled from FUTURE.DO using the same procedure. (If necessary, review Chapter 3.)

Other Kinds of Schedules

The simple format we developed above can be used to record almost anything. Here are some variations.

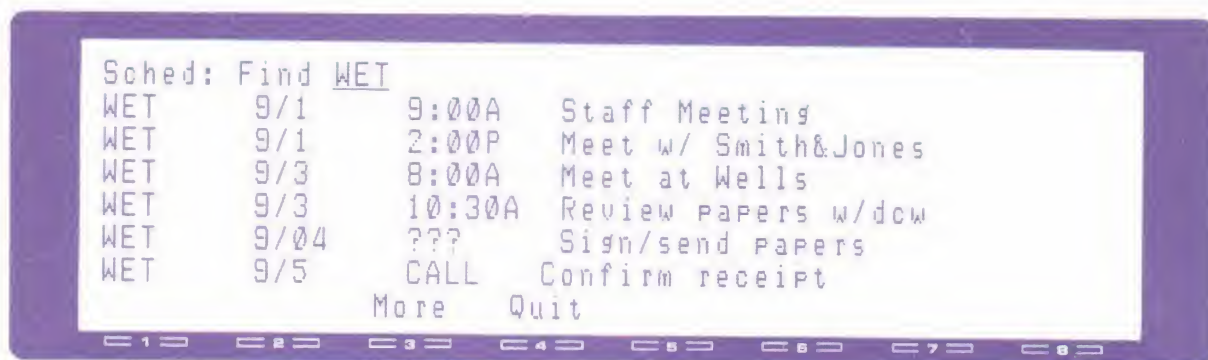
Project Planning

You can use the NOTE.DO file and SCHEDL program for tracking milestones in large projects. In these cases, you might find it useful to add a short “project code” to each line. The example below shows entries for three different projects.

A screenshot of a terminal window with a dark purple border. Inside, a table is displayed with four columns: PRJCT, START DATE, TIME, and DESCRIPTION. The table contains seven rows of data. At the bottom of the terminal window, there is a status bar with eight icons labeled 1 through 8.

PRJCT	START DATE	TIME	DESCRIPTION
TWG	9/1	8:00A	Meet Bob
WET	9/1	9:00A	Staff Meeting
PRO	9/1	12:00P	Lunch with rwl
WET	9/1	2:00P	Meet w/ Smith&Jones
PRO	9/1	4:00P	ACM Board Meeting
TWG	9/2	8:30A	Meet typesetter

Using this format and SCHEDL, you could easily list all the tasks related to project WET, for example, in chronological order. In other words, SCHEDL can be used to view time lines and to help with critical path planning.



You can also use a more formal approach to scheduling several projects. Using this method, lists of tasks pertaining to different projects are entered in individual project files. The most current project file is renamed NOTE.DO, so it can be analyzed through the SCHEDL program. Later, NOTE.DO is renamed back to the project file, and a more current project file can be renamed NOTE.DO.

Cash Flow

Figure- and cash-flow-conscious users will probably want to sacrifice some space from DESCRIPTION and add a column for \$\$\$.\$\$:

DATE	DESCRIPTION	\$\$\$.\$\$
09/01	Meet & bill Bob	+2500.00
09/01	Write checks:	
	Re	-750.00
	Equip Lease	-250.00

If you enter cash flow information in your NOTE.DO file, you can use the SCHEDL program to find all the dollar figures by searching for the decimal point:

```

Sched: Find ,
DATE DESCRIPTION $$$.$$
09/01 Meet & bill Bob +2500.00
      Rent -750.00
      Equip Lease -250.00
09/08 Pay agency -900.00
09/15 Deposit retainer +3000.00

```

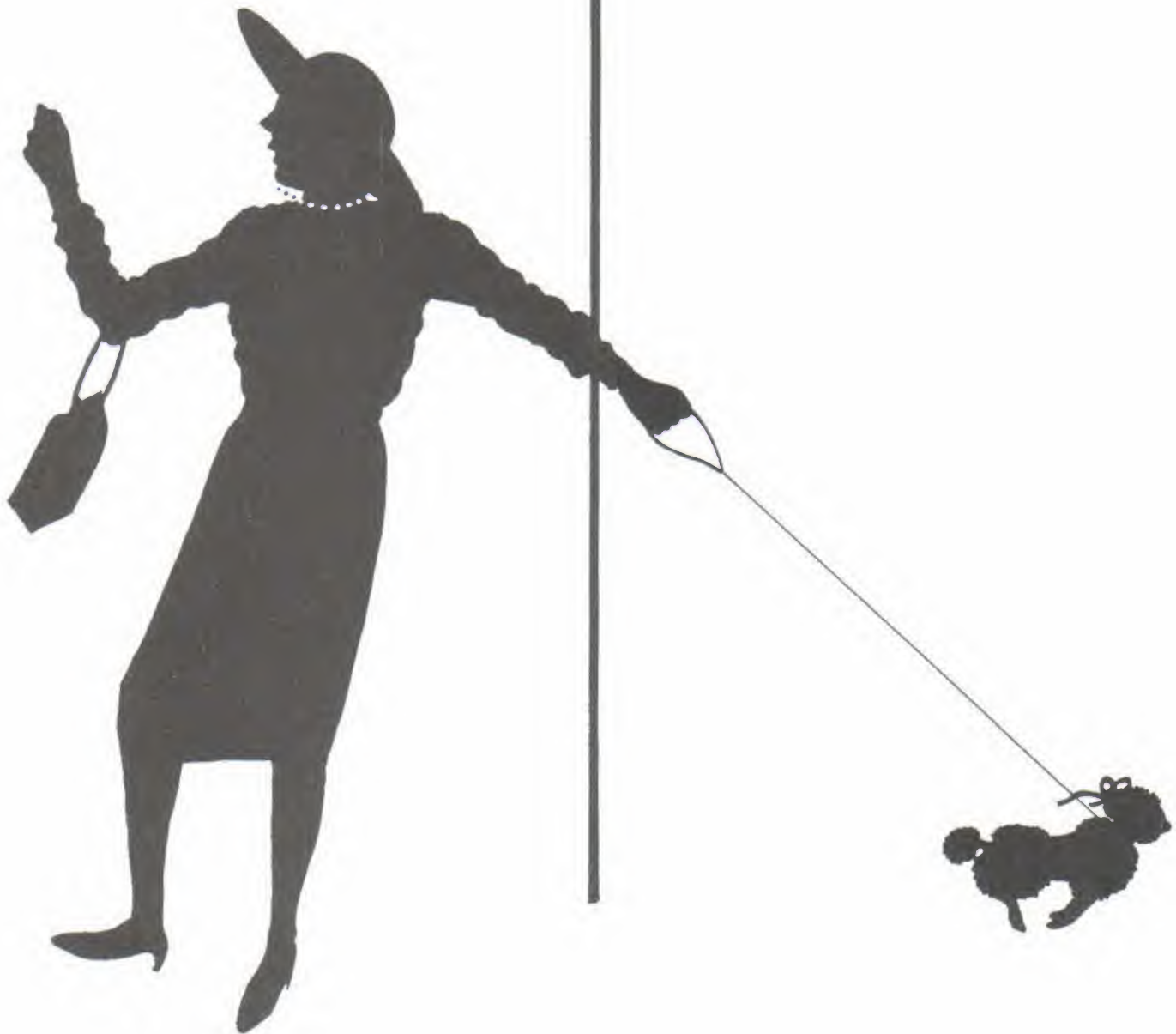

Notepad

You might decide to limit the file by entering only “definite dates” and “firm figures” as in the examples shown above, or you might be like us and fill in anything that comes to mind.

```
09/01 07:30A Drop Tracy - flight UA-232
09/01 ???? Pick up Laundry
09/01 08:00A Meet Bob
09/01 ???? Write checks - including
rent, equipment lease, electric, Jody
09/01 09:00A Staff Meeting - prepare
monthly sales report
???? CALL Smith&Jones 647-1110
09/01 12:00P Lunch with rwl
09/01 ???? Send out contract
```

As you can see, there are many possible uses for a NOTE.DO file. Before you design your own, match your needs against those served by these other formats. Of course, these may be very personal matters for you, and no doubt you will be happy to design your own NOTE.DO file. (For information about printing out the NOTE.DO file, see Chapter 7.)

HARBOP
MARINSHIP



5

More than an ADDRSS Book

Concepts

The ADRS.DO file

A simple ADRS.DO file for use with TELCOM (autodialing)

Using the ADDRSS program

A complex ADRS.DO file for multiple applications

Coding a mailing label program

Coding a program to build your ADRS.DO file

A simple ADRS.DO file for printing mailing labels

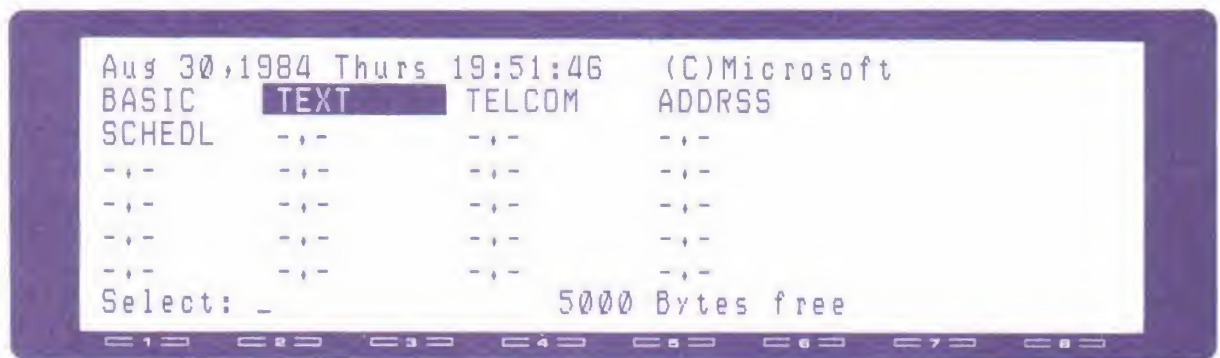
We all know what an address book is, right? It's a little black book filled with names, each of which has an address and/or a phone number associated with it. If your book is anything like ours, it's hardly a model of perfect organization. Obsolete addresses still show under big black X's, and there are so many names beginning with S that the S's have overflowed to the Q page. Loose business cards bulge under a paperclip on the back flap, and one of the pages is about to fall out. Every year or so, the book gets so filled and disorganized that we have to transcribe each address into a new, probably larger, volume. Then we discover "lost" names that were entered under the wrong letter of the alphabet.

You can eliminate many of these inconveniences by using the Model 100 to store your address list. The stored list can be easily and almost instantly searched for a particular name, regardless of the order in which the names are entered. The list can be printed out in hard copy any time you need it, or it can be printed directly onto mailing labels. You can add new names and addresses anywhere in the list and delete obsolete names and addresses without leaving a trace.

As in the scheduling program described in Chapter 4, there are two parts to the Model 100's address-processing capability: the address list, which is a file called `ADRS.DO`, and the program `ADDRSS`. In addition, the `TELCOM` program can also make use of the `ADRS.DO` file, as we'll see. We'll show you how to create the `ADRS.DO` file first and then show you some of the things you can do with it, as well as some of the different formats it's possible to give it for different applications.

The `ADRS.DO` File

To create an `ADRS.DO` file, position the cursor over the word *TEXT* on the main menu and press **ENTER**.



When the screen prompt asks "File to edit?", type `ADRS` and press **ENTER**:

```
File to edit? ADRS
```

Now you can begin typing your address list as you would any text file. Let's start with a simple file containing only names and phone numbers. Later, we'll use more complicated formats.

Type in the following list:

```
ADRS.DO 28 July 1984
Michael Cobb:647-1110
Mainframe:647-9000
Robert Thompson:647-1110
Peter Wilson:821-2121
Julie Zaro:656-8755
```

The first line of this file simply identifies the file and records the date the file was last updated. Then we list the names and corresponding phone numbers. The `TELCOM` program (to be described later) requires that phone numbers be preceded by a colon (:), so we'll follow that convention.

That's all there is to creating the ADRS.DO file. (Of course, a practical file would contain many more numbers than this example.) To do useful work with the ADRS.DO file, you'll need to use either the ADDRSS program or the TELCOM program. We'll first describe how to use ADDRSS.

Using the ADDRSS Program

The Model 100 comes with a built-in program called ADDRSS, which handles some of the most common address list questions. The ADDRSS program will not work until you have entered your address list into the ADRS.DO text file. Once you have created an ADRS.DO file, you can use the ADDRSS program to find a name, or you can ask for a display of all names that begin with "Smi", for example. You can also ask that the name(s) and address(es) be displayed on the screen or printed out on the printer. For example, we can use the ADDRSS program to access the simple phone list used earlier. To begin, position the cursor over the word *ADDRSS* on the main menu and then press **ENTER**.



The screen will change to display the prompt "Adrs:", and there will be new labels for keys **F1** (Find) and **F5** (Lfnd).



Let's try the FIND function first. Press (F1); when the screen displays the prompt "Adrs: Find", type the name you wish to find. In our case, we want to find only those records that include the word *Wilson*:

```
Adrs: Find Wilson
```

If there are no "Wilsons" in the file, the "Adrs:" prompt will be displayed again. Otherwise, all the lines that include the word *Wilson* will be displayed on the screen.

```
Adrs: Find Wilson
Peter Wilson:821-2121
Adrs:
```

If we press (F1) and then press (ENTER) without specifying what name to search for, the ADDRSS program will find *everything*; that is, every entry in the ADRS.DO file will be displayed.

```
Adrs: Find
ADRS.DO 28 July 1984
Michael Cobb:647-1110
Mainframe:647-9000
Robert Thompson:647-1110
Peter Wilson:821-2121
Julie Zaro:656-8755
```

If your Model 100 is already hooked up to a printer, you can press (F5) (ENTER) to print out the entire file. (Caution: if your Model 100 is *not* hooked up to a printer, then pressing (F5) may lock the keyboard, and you will need to press RESET to regain control.)

Using TELCOM with the ADRS.DO File

As we noted earlier, the Model 100 comes with two built-in programs that work with ADRS.DO — TELCOM and ADDRSS. Through TELCOM, you can look up a number in your ADRS.DO file and then have the Model 100 automatically dial the number through your phone line. You can use TELCOM to dial a number and then pick up the phone for direct conversation, or you can use it to dial another computer and use your Model 100 to send or receive information.

This section illustrates the use of TELCOM with the simplified ADRS.DO file from the last section, which includes only names and phone numbers. Of course, the same TELCOM commands will work with the more complex ADRS.DO file shown later in this chapter.

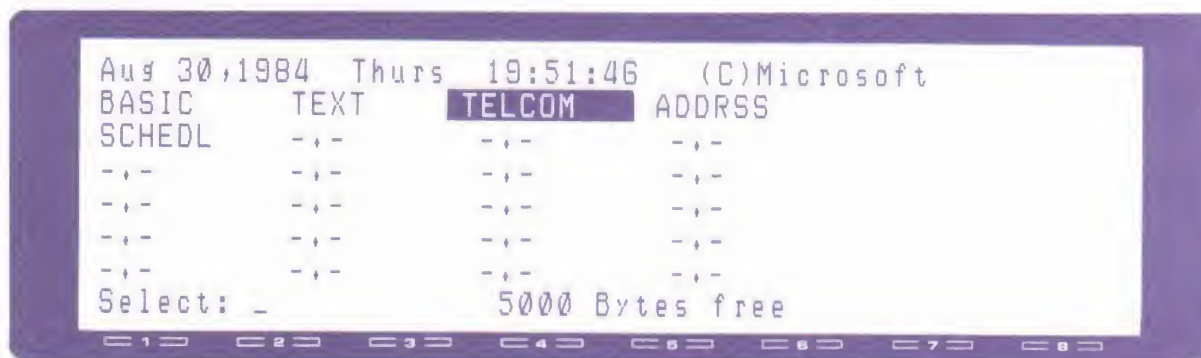
Here's the phone list `ADRS.DO` file that we used in the last section:

```
ADRS.DO 28 July 1984
Michael Cobb:647-1110
Mainframe:647-9000
Robert Thompson:647-1110
Peter Wilson:821-2121
Julie Zaro:656-8755
```

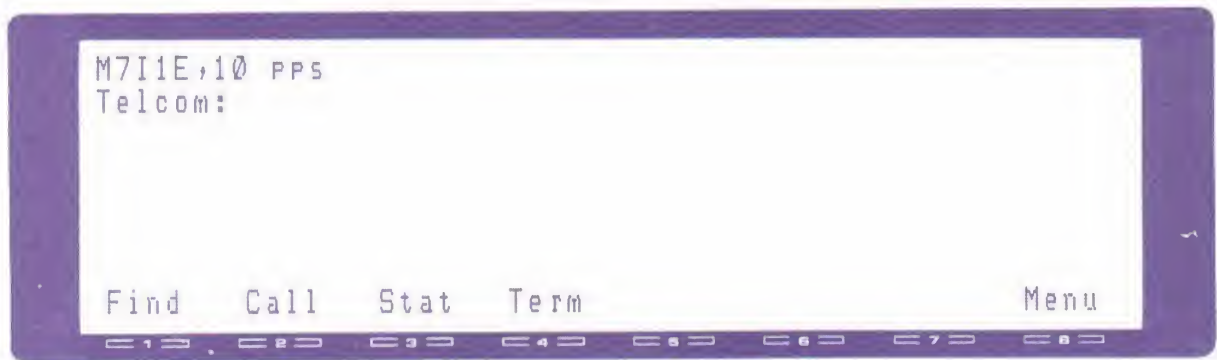
The essential requirements for using the autodialing feature of `TELCOM` are the following:

- The phone number must be entered on the same line as the name, that is, before any carriage return.
- The phone number must be preceded by a colon; that is, no other colons may appear in the same record.
- The phone number may be entered with parentheses around the area code and a hyphen following the exchange or as a series of numbers only; the area code is optional.

If your `ADRS.DO` file meets these requirements, you can try the auto-dial feature of `TELCOM` by following these instructions. Begin by hooking the Model 100 up to your phone line (as described in Chapter 9); then position the cursor over the word *TELCOM* on the main menu and press **ENTER**.



The top line of the screen changes to show the `TELCOM` status settings. (We discuss these in detail in Chapter 9; if your top line does not show "M711E,10", then you should read that chapter now.) The second line shows the "Telcom:" prompt. The function key labels show your options: Find, Call, Stat, Term, and Menu.



To find a name and number in your ADRS.DO file, press **(F1)**, type the name you are seeking (or part of the name), and press **(ENTER)**.



The labels on the screen's bottom line show you that you can *call* the found number by pressing **(F2)**, search for more Wilsons by pressing **(F3)**, or end the search-autodial process by pressing **(F4)**. In this case, with the Model 100 connected to our phone lines, we press **(F2)**. TELCOM automatically dials the found number.



TELCOM ignores parentheses and hyphens in the phone number and simply dials the numbers that follow the colon in the found record, in sequence. Once the number is dialed, you can pick up the phone and

converse with the human at the other end of the line, or you can press **(F3)** (Stat) to begin communications with the computer at the other end of the line. (See Chapters 9 and 11 for more details about communicating with other computers.)

More Complex ADRS.DO Files

Of course, your ADRS.DO file can include more than just names and telephone numbers. You may have company names, titles, full addresses — even birthdays. Let's see how ADRS.DO works with more complex files by building an ADRS.DO file that can be used with a variety of applications.

- To use the ADDRSS program effectively, each full name-address-phone record must be entered on *one* line only, without carriage returns.
- To use TELCOM for autodialing, the phone number must be preceded by a colon, and no other colons may appear in the same record.
- To use the mailing list program shown later in this chapter, the items within each record must be typed in the same order — name, address, phone — and elements of each record must be separated by a comma:

```
firstname,lastname,company,address,city,state,zip,:phone,
```

- Furthermore, to use the mailing list program, each record must contain the same number of commas or separators. For example, a comma is included for the company name, even if no company name is entered. Also note that the first record (the ADRS.DO header line) has commas for all but the first field.

```
firstname,lastname,company,address,city,state,zip,:phone,  
firstname,lastname,,address,city,state,zip,:phone,  
,,company,address,city,state,zip,:phone,
```

Here's an example of this kind of ADRS.DO file:

```
ADRS.DO 28 July 1984,,,,,  
Michael,Cobb,Major Enterprises,2300 Main  
Street,Suite 13,Fairview,CA,94110,:647-1110  
,,Mainframe,,,,,:647-9000      ← Record for TELCOM only  
Robert,Thompson,Major Enterprises,2300  
Main Street,Suite  
99,Fairview,CA,94110,:647-1110
```

```
Peter,Wilson,More Major  
Holdings,Penthouse,3434 First  
Avenue,Fairview,CA,94110,:821-2121  
Julie,Zaro,,1543 Park  
Boulevard,# 305,Fairview,CA,94110,:656-8755
```

← Record has no company name

Although the information for each record has been entered as one line, the Model 100 screen display will break the lines at spaces between commas or separators.

This file format may seem hard to read at first, but it can be read easily by the ADDRSS and TELCOM programs. In addition, you can write your own BASIC programs to read the file and print it out in whatever form you wish. (In the next section, we will give an example of a BASIC program that will print the file shown above onto mailing labels, omitting the phone number.)

If you haven't already, start an ADRS.DO file using TEXT and enter a few addresses of your own, or type in the first two addresses shown in the example below, putting phone numbers at the end of each record. (Later in this chapter, we'll show you a short and simple BASIC program that will help you build this file more easily.) If you have already entered the phone list shown in the last sections, then you can simply edit that file by adding an address to each record — remembering to include commas in the right places.

Once you have used TEXT to type in a file like the one shown, return to the main menu by pressing **(F8)**; then position the cursor over the word ADDRSS and press **(ENTER)**. The screen will change to display the prompt "Adrs:" — and there will be new labels for keys **(F1)**, "Find", and **(F5)**, "Lfnd". Let's try the FIND function first. Press **(F1)**; when the screen displays the prompt "Adrs: Find", type the name you wish to find. In our case, as in the last section, we want to find only those records that include the word *Wilson*:

```
Adrs: Find Wilson
```

If there are no "Wilsons" in the file, the "Adrs:" prompt will be displayed again; otherwise, all the lines that include the word *Wilson* will be displayed on the screen.

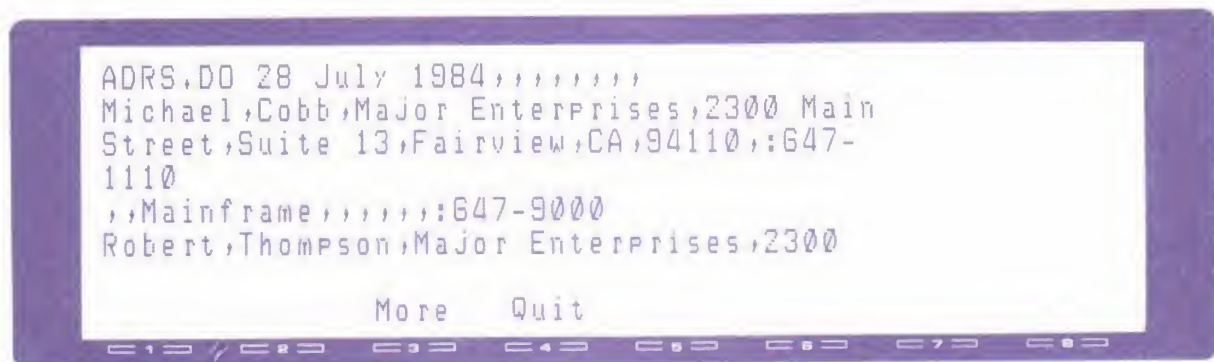
```
Adrs: Find Wilson  
Peter Wilson,More Major  
Holdings,Penthouse,3434 First  
Avenue,Fairview,CA,94110,:821-2121  
Adrs:
```


The full record for Mr. Wilson is displayed, because it was typed in as one line — we did not press **(ENTER)** until we finished typing the phone number. If we had pressed **(ENTER)** after typing the name in order to put the company name on a new line, then the ADDRSS program would *not* be able to FIND both the name and the address at the same time.

If the list requires more than six lines on the screen, then the first six lines will be shown, and you may view more by pressing **(F3)**, “More”. To see how this works, press **(F1)** again to display the “Find” prompt; then just press **(ENTER)**.

Adrs: Find

This is equivalent to saying “Find *everything* in the ADRS.DO file”. The screen displays the first six lines of the ADRS.DO file, and new labels are shown for **(F3)**, “More”, and **(F4)**, “Quit”.



To view more lines of the file, press **(F3)**; to end the “Find” process, press **(F4)**. When you press **(F4)**, or when you reach the end of the ADRS.DO file, the “Adrs:” prompt appears again.

You can use the LFND function to print this file out, but, as you can guess, the printed version will not be suitable for mailing labels, because every record is typed on *one* line, including a lot of commas! In the next section, we provide an example of a simple BASIC program you can use to print this file out to mailing labels. (For more examples of printing using the ADDRSS program, see Chapter 7.)

A BASIC Mailing Label Program

Here is a program you can type in and use to print one-inch mailing labels for any file arranged in the order shown, with each element of the record separated by commas:

```
firstname,lastname,company,address1,address2,city,state,zip,:phone
```

Each of these elements is given a name in BASIC (see line 40 of the program below). (For details on coding BASIC programs, see Chapter 6.)

Because commas are used in a special way in this file, no additional commas may be entered. For example, do *not* include commas in the company name. Major Enterprises is a valid company name entry, but Major Enterprises, Inc. is not valid because it includes a comma.

To use this program, position the cursor over the word *BASIC* on the main menu, then type each line shown below. Be sure to include a line number at the beginning and to press **ENTER** at the end of each line.

```

10 REM MAILST ENTER
12 CLEAR 2000 ENTER
15 DIM FN$(100),LN$(100),CO$(100),S1$(100),S2$(100),C$(100),
ST$(100),ZP$(100),PH$(100) ENTER
20 OPEN "RAM:ADRS.DO" FOR INPUT AS 1 ENTER
30 FOR N=1 TO 100 ENTER
40 INPUT #1,FN$(N),LN$(N),CO$(N),S1$(N),
S2$(N),C$(N),ST$(N),ZP$(N),PH$(N) ENTER
50 IF EOF(1) THEN G0 ELSE NEXT ENTER
60 CLOSE ENTER
70 FOR I=1 TO N ENTER
80 LPRINT FN$(I) " " LN$(I) ENTER
90 LPRINT CO$(I) ENTER
100 LPRINT S1$(I) ENTER
110 LPRINT S2$(I) ENTER
120 LPRINT C$(I) " " ST$(I) " " ZP$(I) ENTER
130 LPRINT " " : NEXT ENTER
999 MENU ENTER

```

← The maximum number of addresses = 100

As given, the program allows for up to 100 records. If you have 8K or less available bytes, you will need to make the number in parentheses in line 15 smaller, as well as the number following the word TO in line 30. Try 50, and if you still get an OM (Out of Memory) error, try a smaller number like 25 or 30. Also, you may run out of string storage space if you process many records. If you get an OS (Out of String space) error when running the program, you can increase the number after CLEAR in line 12. This is also limited by your memory, however, so it's best to have more than 8K available for running this program.

For example, to modify this program to handle up to 50 records, edit lines 15 and 30 to read:

```

15 DIM FN$(50),LN$(50),CO$(50),S1$(50),S2$(50),C$(50),ST$(50),
ZP$(50),PH$(50)
30 FOR N=1 TO 50

```


To run this program, press **(F4)** — but remember, you must have already created an ADRS.DO file like the one shown in the previous section, and your Model 100 must be hooked up to the printer!

Running this program on the file shown in the previous section yields labels like those shown below. Notice that there are blank lines in records where the full address was not entered in the ADRS.DO file.

ADRS.DO 28 July 1984

Michael Cobb
Major Enterprises
2300 Main Street
Suite 13
Fairview, CA 94110

Mainframe

Robert Thompson
Major Enterprises
2300 Main Street
Suite 99
Fairview, CA 94110

Peter Wilson
More Major Holdings
Penthouse
3434 First Avenue
Fairview, CA 94110

Julie Zaro

1543 Park Boulevard
305
Fairview, CA 94110

Once this program works the way you expect it to, you can save it under its own filename by pressing **(F3)** in BASIC and typing the filename MAILST.

TRS-80 Model 100 Software
Corp, 1983 Microsoft
2821 Bytes free
Ok
Save "MAILLST
Ok

A BASIC Program to Build Your Address File

If you find that entering names and addresses into a "complex" ADRS.DO file is inconvenient because of all the required commas, you can type in a BASIC program to make entry easy. The program shown below can be used to create the ADRS.DO file that we printed out in the previous section.

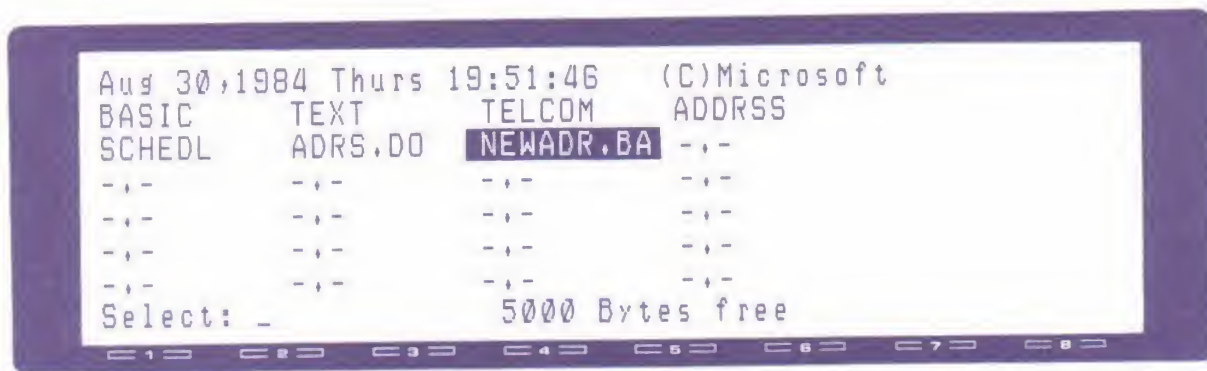
Get into BASIC as shown in the last section and type in the program .

```
10 REM NEWADR
20 OPEN "RAM:ADRS.DO" FOR APPEND AS 1
30 INPUT"FIRST NAME";FN$
40 INPUT"LAST NAME";LN$
50 INPUT"COMPANY";CO$
60 INPUT"FIRST ADDRESS LINE";S1$
70 INPUT"SECOND ADDRESS LINE";S2$
80 INPUT"CITY";C$
90 INPUT"STATE";ST$
100 INPUT"ZIP";ZP$
110 INPUT"PHONE";PH$
120 PRINT #1,FN$," ",LN$," ",CO$," ",S1$," ",S2$," ",C$," ",ST$," ",ZP$," ",PH$
130 CLOSE
140 INPUT"MORE (Y/N)";A$
150 IF A$="Y" THEN 20 ELSE 160
160 IF A$="Y" THEN 20 ELSE 999
999 MENU
```

Sample Run

If you type in the NEWADR program in BASIC and save it under the name NEWADR.BA, then you can run it at any time to add new names and addresses to your ADRS.DO file. To *change* addresses that have already been entered, edit ADRS.DO directly as a text file using TEXT.

To run the program, position the cursor over the word NEWADR on the main menu and press **ENTER**. (The file named ADRS.DO must already exist.)



The NEWADR program will begin immediately, prompting you for entries into the ADRS.DO file. Remember to press **ENTER** after each entry.

```
FIRST NAME? Mildred
LAST NAME? Pierce
COMPANY? Pierce's Diner
FIRST ADDRESS LINE? 120 Boardwalk
SECOND ADDRESS LINE? SPACE
CITY? Hollywood
STATE? NY
ZIP? 20202
PHONE? :(212)634-5789
MORE (Y/N)? N
OK
```

This new information will now be added to the ADRS.DO file. Notice two peculiarities in this program:

- In making a series of entries, you can “ditto” information from the previous record by pressing **ENTER** without entering data when the prompt appears (that is, STATE).
- Otherwise, when there is no data for a particular field (such as SECOND ADDRESS LINE), you must at least press the space bar before pressing **ENTER**. Also note that you need to include the colon before the phone number. If you run MAILST.BA again, you'll see that the format of the labels for the records added with NEWADR.BA is a little different. If you learn more about BASIC, you can modify it to suit yourself.

This program is an example of what can be done for any structured ADRS.DO file. If you decide to use this program or to code one for yourself, be sure to read Chapter 6. For a complete course in programming in BASIC on the Model 100, see *Mastering BASIC on the TRS-80® Model 100*, by Bernd Enders (New York: Plume/Waite, New American Library, 1984).

A Simple ADRS.DO File for Printing Mailing Labels

As you can see, the format you choose for your ADRS.DO file affects the ways you can use the file. A list entered for printing mailing labels will not be useful for ADDRSS or TELCOM; a list entered for ADDRSS and TELCOM may need a special program in order to print out in mailing list format. In this section, we review a simple file that can be used to print mailing labels — but cannot be used by ADDRSS or TELCOM.

```
Michael Cobb  
Major Enterprises  
2300 Main Street  
Suite 13  
Fairview, CA 94110
```

```
Robert Thompson  
Major Enterprises  
2300 Main Street  
Suite 99  
Fairview, CA 94110
```

```
Peter Wilson  
More Major Holdings  
Penthouse  
3434 First Avenue  
Fairview, CA 94110
```

This file format has the advantage of being easy to read and ready to go for printing. The only disadvantage is that it will not work with the built-in ADDRSS and TELCOM programs. If you want to get the most out of your ADRS.DO file — that is, if you want to be able to search for full names and addresses and to autodial through the modem — then this is not the format for you. We offer it here as an example of “common-sense” formatting with limited applications.

All we need to remember is that each name and address must be five lines long, followed by one blank line before the next address, in order to fit exactly on one-inch labels. If an address is only four lines long, then it must be followed by two blank lines in order to maintain the right spacing on the printed labels.

Once you have typed a few addresses, you can try printing them out directly by pressing **CTRL** Y. You can print only the address showing on the screen by pressing **PRINT** (see Chapter 7).

With this type of file, the ADDRSS program will not be able to FIND full name-and-address records, because each record is broken up by car-

riage returns. Instead, you can use the FIND function in the TEXT program. To find a particular name and address, press **(F1)**. When the TEXT program prompts "String:", type the name you seek and press **(ENTER)**. For example:

```
String: Wilson
```

The cursor jumps to the first place "Wilson" appears in the address list, and you can see the address on the lines that follow.

```
Suite 99  
Fairview, CA 94110
```

```
Peter Wilson  
More Major Holdings  
Penthouse  
3434 First Avenue  
Fairview, CA 94110
```

If the first "Wilson" found is not the one you wanted, you can search for the next one by pressing **(F1)** again. The prompt will appear with the word "Wilson" already entered, and you just press **(ENTER)** to search again. When all the Wilsons are found (or if *none* are found), the Model 100 will display the message "No match" on the last line of the screen.

Of course, we can print this file or search it just as we can print or search *any* text file, but this file cannot take advantage of the built-in programs ADDRSS and TELCOM or of your own BASIC programs. Why? Because ADDRSS, TELCOM, and BASIC programs require something in the ADRS.DO file that this example does not have.

- The ADDRSS program expects each full name-and-address record to be entered on *one* line only, without carriage returns. In our file, we have a carriage return after the name and each line of the address.
- The TELCOM program expects the phone number to be somewhere on the same line as the name, and the phone number must begin with a colon (:). In our file, we have not even considered phone numbers.
- BASIC programs expect each full name-and-address record to be entered on one line, with the same sequence of items within each record and each item separated by commas (or some other common separator).

So why use this format? It is simple to enter and it prints mailing labels easily.



6

BASIC Basics

Concepts

- Creating a program
- Storing a program
- Running a program
- Editing a program
- Renaming a file
- Deleting a file

You needn't be a technically skilled programmer to learn a few simple tricks with Microsoft BASIC on the Model 100. BASIC is not only a programming language, it is also a simple way to tell the Model 100 what to do. For example, you can use BASIC to change the names of files or to erase files. With BASIC, you can also copy your files from the Model 100 memory onto cassette tape or load files from cassette tape into memory.

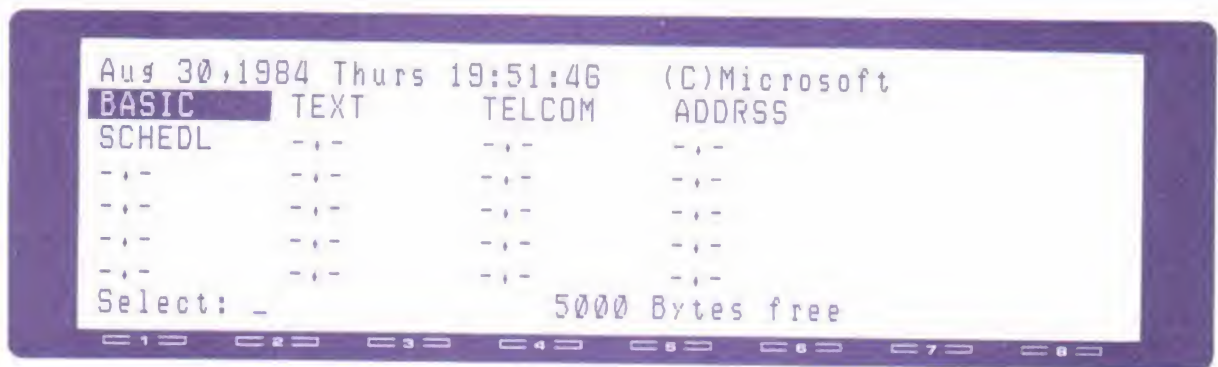
In this chapter, we don't intend to teach you BASIC programming. Instead, we'll show you some short, simple programs that you can type in and use. Using these examples, you will learn the steps required to create, save, run, edit, copy, delete, and load *any* BASIC program. You can follow the same steps when you type in other BASIC programs (such as those in Chapter 5).

All BASIC programs have certain steps in common. First, you must enter BASIC and type in the various commands that make up the program. Usually, you will run the program at least once before deciding to save it. After saving the program, you can run it again, edit it, rename it, send it to another computer, or save it onto cassette tape. Finally, there may come a time when you wish to delete the program from the Model 100's memory.

Some BASIC programs will use other files in addition to the program file itself. For example, one program may cause another program to run, or a program may be written specifically to read or print a text file.

Creating a Program

To enter BASIC, position the cursor over the word *BASIC* on the main menu and press **ENTER**.



You are now in BASIC, and the Microsoft copyright message will appear on the screen of your Model 100, along with the number of bytes available for BASIC's use (this will vary, depending on the size of your memory and how much you have already stored).



Now that you are in BASIC, you can type in *direct commands*, which will be executed immediately, or *numbered instructions*, which you can save and execute whenever you wish. One or more numbered instructions constitute a BASIC *program*.

Direct commands (without line numbers) are executed immediately when you press **ENTER**. You learned two direct commands in Chapter 2 when you set the date and time. Remember that we set the date by entering BASIC from the main menu and then typing the following:

```
DATE$="09/01/84"
```

When you type the line above, the Model 100 obeys the command immediately and sets the internal date as September 1, 1984. Other common direct commands include SAVE, LOAD, RENAME, and KILL, examples of which are given later in this chapter.

You can also use direct commands to perform simple calculations. For example, you could divide a \$15.78 lunch bill in half by typing the following:

```
PRINT 15.78/2
7.89
OK
```

The screen immediately displays the results of this calculation: 7.89. Direct commands like this cannot be saved; once you press **ENTER**, they are executed and forgotten. To run the same command again, you must retype it and press **ENTER** again.

To save a direct command as a program, you must enter it on a numbered line. Unlike direct commands, which are executed as soon as you press **ENTER**, numbered lines will not be executed until you run the program. Furthermore, whereas direct commands are typed in one at a time and then forgotten, any number of numbered lines (up to memory capacity) can be typed and saved in the computer's memory.

For example, let's say that you frequently have lunch with others and divide the check evenly among you. You could just type the PRINT command, followed by the check amount, a slash, and the number of people present (as we did above), or you could type and save a program that would make this easier for you. Here's how. Each numbered instruction begins with a line number in column 1 of the screen and ends with a carriage return (the **ENTER** key). To create our LUNCH program, for example, we would type the following lines, pressing **ENTER** after each one:

```
10 PRINT"LUNCH DIVISION"
20 INPUT"Check Total";CT
30 INPUT"Number of parties present";PT
40 EA=CT/PT
50 PRINT"Amount due from each ="EA
```

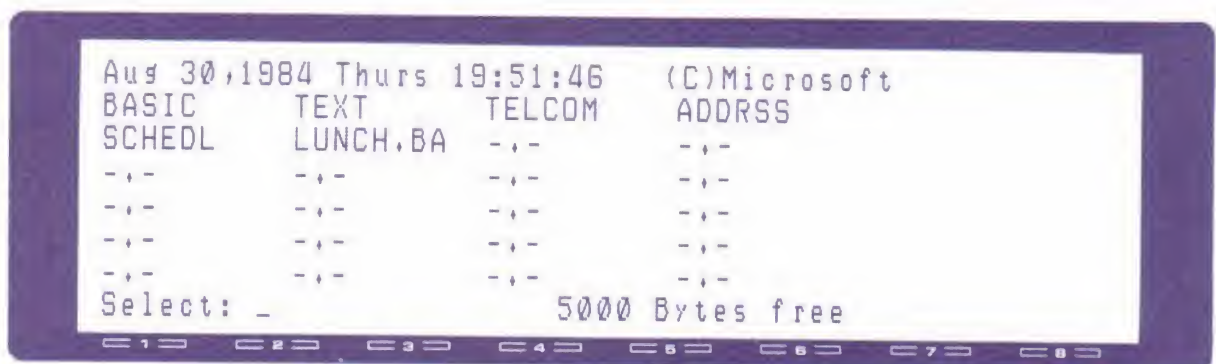
Now, to RUN the program, press (F4). The screen will display two questions. Answer and press (ENTER) after each to find out how much each person owes:

```
LUNCH DIVISION
Check Total? 15.78
Number of Parties Present? 2
Amount due from each = 7.89
OK
```

When the program is finished, the Ok prompt is displayed on the screen. You can view the list of lines you have typed by pressing (F5) and LIST. Before you type another program in BASIC, it's a good idea to SAVE this program under its own filename. To do this, press (F3) (SAVE), type the name of the new file, and then press (ENTER). For example:

Save "LUNCH"

Now press (F8) to return to the main menu, and you will see the new filename LUNCH.BA listed on the screen.



If you have just entered BASIC for the first time or have saved a program in a previous session under its own filename, you can assume that there is nothing stored in the current BASIC memory area. You can type a new program.

If, however, you suspect that there is something left in the memory from a previous session, you can view it by entering BASIC and pressing (F5), "List". If there are numbered lines in the BASIC memory area, you can RUN or EDIT them, SAVE them under their own filename, or erase them completely by typing NEW and pressing (ENTER).

NEW

Here's another kind of program to try. Enter BASIC from the main menu, press **(F5)** to be sure that no unsaved program lines are in the BASIC memory area, and then type the following:

```
10 CLS
20 FOR N=1 TO 60
30 PSET(N,N)
40 SOUND 100*N,4
50 NEXT
```

The lines above will cause the Model 100 to make *sound* and draw a line on the screen — try it! Numbered instructions will be executed in numerical sequence, regardless of the order in which they were typed.

As you progress with your programming experience, you will find many occasions for listing a program. Perhaps you will run it for a friend, who will then beg you to show him every line of clever instruction. Perhaps you will want to remember what you have already coded before you add more lines of instruction. You have forgotten to type a comma or have made some other programming error and will want to view the program before you start editing lines.

To list the numbered lines of the current program, press **(F5)**. The numbered lines that you have typed in BASIC will be displayed in numerical sequence on the screen. If the program is longer than eight lines, then displayed lines will scroll off the top of the screen as more lines scroll up from the bottom. To pause this scrolling process and view the displayed lines, press **(PAUSE)**. To resume the display of more lines, press **(PAUSE)** again. To stop the LIST command, press **(SHIFT) (BREAK)**.

To list part of a program, type LIST and a range of line numbers, and then press **(ENTER)**. For example, to list line 10 only, type:

```
LIST 10
```

To list lines 10-30, type

```
LIST 10-30
```

The Function Keys — Common BASIC Commands

Five function keys on the Model 100 are programmed to type in the most common programming functions:

- (F1) Files Lists the names of the files stored in memory.
- (F2) Load“ Prompts for the name of a program to be loaded. Simply type the name of a saved file and press (ENTER). To cause the program to start running as soon as it is loaded, type: Load“filename”,R.
- (F3) Save“ Prompts for the name under which to save current program. You can save a copy of the program as a TEXT file by typing: Save“filename”,A.
- (F4) Run Causes current program to execute.
- (F5) List Causes the lines of the current program to be listed on the screen. Press (PAUSE) to interrupt the scrolling of lines off the top of the screen, then (PAUSE) again to resume scrolling.

Press (SHIFT) (BREAK) or (CTRL) C to terminate a running program.

Storing a Program

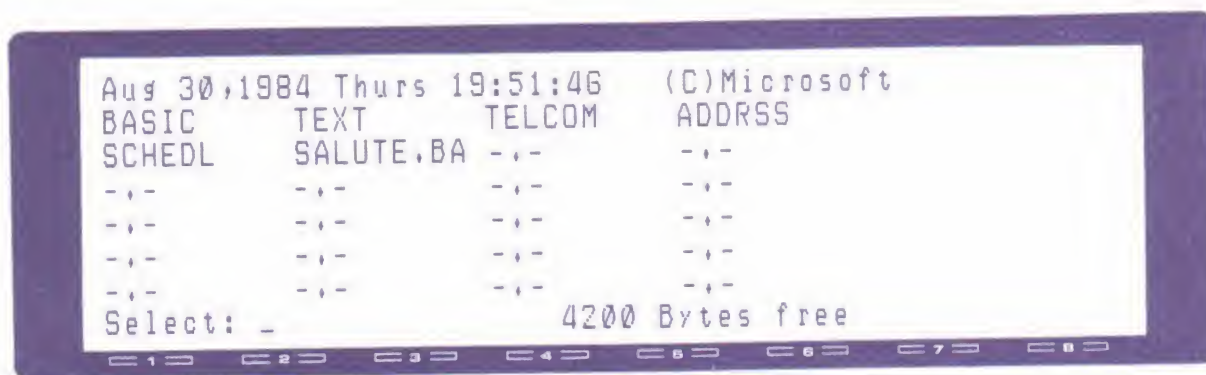
Once you have typed a program consisting of numbered lines of instruction in BASIC, those lines will remain in the BASIC memory until you delete them. You can press (F8) to return to the main menu, and when you return (by positioning the cursor over the word BASIC and pressing (ENTER)), your program will still be there. You can press (F5) to view it.

If you turn off the Model 100, the numbered lines will still remain in the BASIC memory. So why save the program? First, you cannot begin a new program until the old one is saved in its own file and deleted from the BASIC memory. Second, whatever is sitting in the BASIC memory will be erased when you use the LOAD command.

To save the contents of the BASIC memory, press **(F3)** (Save). The screen will prompt you for the filename. Enter a filename no longer than six characters and beginning with a letter of the alphabet. For example (the second quote mark is optional):

Save "SALUTE"

In a moment, the screen replies "Ok". Now press **(F8)** to return to the main menu, where the new file SALUTE.BA is listed:



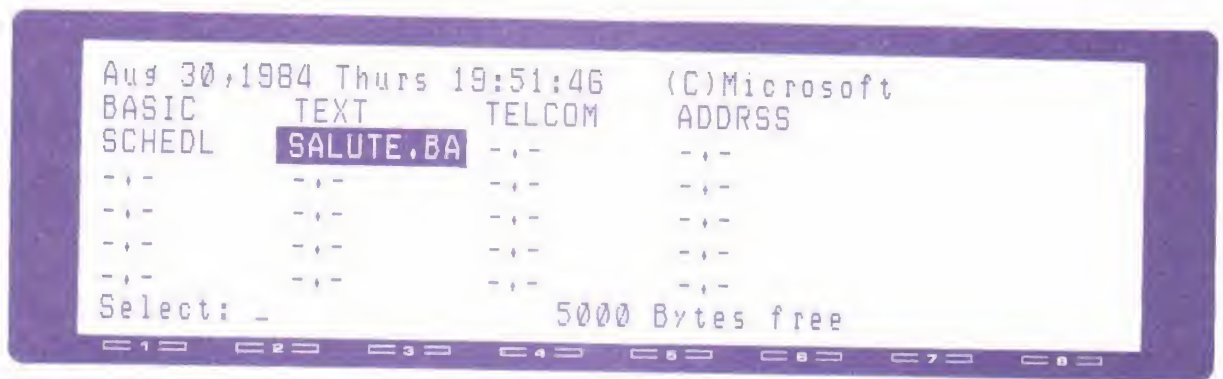
All BASIC programs saved as files on the Model 100 will automatically be given the suffix ".BA" after the name you choose. The name itself may have as many as six characters.

Once you have saved a program, you can return at any time to run, edit, or rename it, send it to another device, or delete it completely from memory (RAM).

Running a Program

As already mentioned, you can always run the program you are typing under BASIC by pressing **(F4)** and RUN. Once you have saved the program as a file, there are three ways to run it.

From the main menu, you can run a program by positioning the cursor over the filename and pressing **(ENTER)**.



The program will automatically begin to execute as soon as you press **(ENTER)**. When the program is finished, "Ok" will be displayed on the screen. Now you can use the F keys to run the program again or list the program. (You can also edit the program — see below.) When you press **(F8)** and return to the main menu, the program is saved automatically.

If you are in BASIC, you can run a program you previously saved under another filename by typing RUN and the name of the program. For example:

```
RUN "SALUTE"
```

You can enter BASIC from the main menu, then load an existing file by pressing **(F2)** and typing the name of the program. For example:

```
Load "SALUTE"
```

After a brief pause, the screen will display "Ok", and you can then run, list, or edit the program.

You can code a numbered instruction in one program to execute another program. For example:

```
10 INPUT "What is the date";DATE$
20 INPUT "What is the time";TIME$
30 RUN "SALUTE"           ← This will cause SALUTE to be executed
```

If you have already saved the program file SALUTE.BA, you can enter BASIC from the main menu, type the three lines shown above, and press **(F4)** "RUN".

Finally, you can cause a program to run automatically every time you turn the Model 100 on by using the IPL (Initial Program Load) command. For example, you can enter BASIC from the main menu and type

```
IPL "SALUTE.BA"
```


Now, whenever you turn on your Model 100, it will greet you with the short tune and graphic created by the SALUTE.BA program.

Editing a Program

Even the most perfect programmer runs into program errors. The Radio Shack *TRS-80® Model 100 Portable Computer* manual lists 32 possible error messages. The most common one for beginners is probably SN — a syntax error caused by missing punctuation or some other typographical error. The error message displayed on the screen gives the number of the line that caused the problem, so you can LIST that line and EDIT it appropriately.

Editing a BASIC program is different from editing a text file, because text files and BASIC programs are not stored the same way in memory. Simply stated, text files are stored character for character, while BASIC programs are stored in a condensed form. Generally, you will find it easier to edit short lines by simply retyping them. The EDIT Mode in BASIC is more useful to edit long lines, several lines at once, or even a whole program.

BASIC Editing Rules

- Instructions typed without a line number are executed immediately; they are not saved in memory.
- Instructions that begin with a new line number are added to the other numbered instructions in the memory.
- To replace a line, type the line number and the new instruction. When you press **ENTER**, the new line will replace the old line of the same number.
- To edit a line without retyping the whole thing, type EDIT and the line number, then press **ENTER**. You can use the arrow keys to move the cursor along the line and make corrections. When you have completed the edit, press **F8**.

Under this last option, the BASIC instruction is converted to TEXT. You can use all the methods of editing described for text (see Chapter 3). When you press **F8**, the edited lines of text are converted to code again.

You can EDIT one line, a group of lines, or the whole program at once. For example:

EDIT 10 ← Edits line 10

or

EDIT 10-30 ← Edits lines 10-30

or

EDIT ← Edits entire program

When you finish making changes in EDIT Mode, press **F8** to return to BASIC Mode. The Model 100 will flash the message "Wait" on the screen while it converts the lines from text to BASIC again. If you EDIT large programs all at once, this waiting period can last several minutes, so it's better to edit only a line or two at a time.

Renaming a File

The NAME command is a direct command that changes the name of a file without affecting the file's contents. For example, to change the filename SALUTE.BA to SOUNDS.BA, type

```
NAME "SALUTE.BA" AS "SOUNDS.BA"
```

The NAME command can be used to rename TEXT files as well; for example:

```
NAME "NOTE.DO" AS "DONE.DO"
```

Deleting a File

To clear the current BASIC memory and begin typing a new program, simply enter BASIC from the main menu and type

```
NEW
```


Use the KILL command to delete named BASIC and TEXT files from memory (RAM). You must enter the full filename, including the suffix. For example:

```
KILL "SALUTE.BA"
```

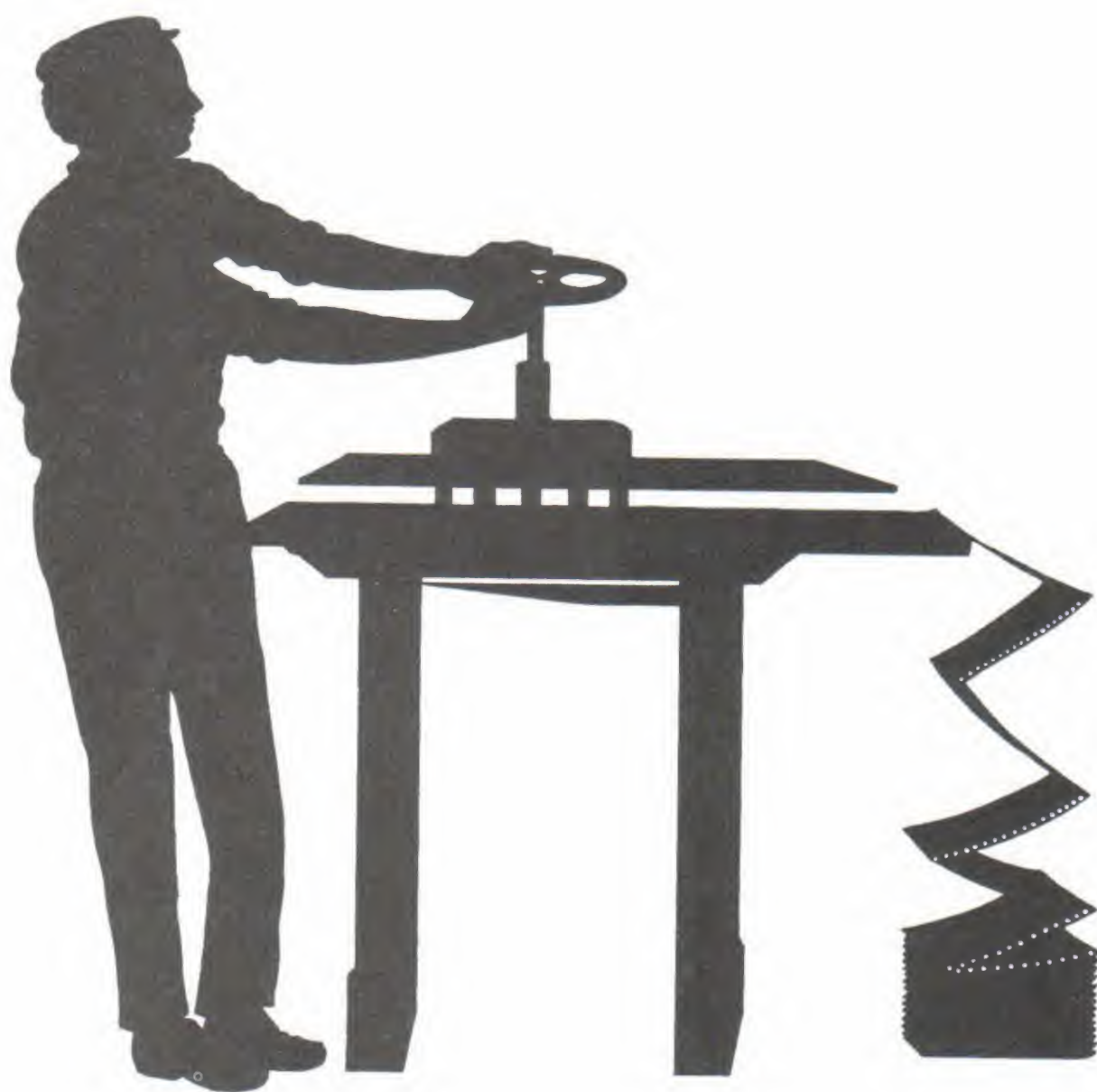
Once a file is deleted, of course, you cannot run, edit, print, or do anything else with it. You'll probably want to print the file, save it on cassette, or send it to another computer before you use the KILL command. (For information about saving files on cassette tape, see Chapter 8; to learn about sending files to another computer, see Chapter 11.)

On Your Own

Here's a program that you can type in, run, edit, and save on your own. We saved our version under the program name ADD.BA. This program turns your Model 100 into a simple adding machine. It will prompt you for a series of values; entered numbers will be added to the accumulating total — values preceded by a minus sign (a hyphen) will be subtracted. It's as simple as that!

Whenever you answer the prompt VALUE? by just pressing **ENTER** without typing a numeric value, the program will display the subtotal of your current entries, along with the accumulating grand total, which includes previous subtotals.

```
10 REM ADD.BA
20 T=0
30 CLS:S=0:PRINT"SIMPLE ADDITION (AND -SUBTRACTION)"
40 PRINT"(TO GET TOTAL, PRESS ENTER ONLY)"
50 V=0
60 INPUT"VALUE";V:S=S+V
70 IF V=0 THEN 80 ELSE 50
80 PRINT USING" SUBTOTAL = #####,##";S
90 T=T+S:PRINT USING"GRAND TOTAL = #####,##";T
100 INPUT"ENTER 1-ADD MORE 2-CLEAR/RETRY 9-EXIT";A
110 ON A GOTO 30,20,,,,,999
999 MENU
```



7

Printing Files

Concepts

What you need

How to hook up the Model 100 to a printer

Printing the screen image: **PRINT** key

Printing whole text files: **CTRL** Y

Printing BASIC programs: LLIST

Printing BASIC programs: SAVE "LPT:"

A BASIC print program: L **PRINT**

Printing in SCHEDL and ADDRSS: **F4**

Some say the day is fast approaching when all our text will be online, and readers will view everything they need to read through video terminals. The latest news, and even novels and poetry, will be as accessible as telephones are today. In short, paper will be obsolete as a medium for communication.

But, until that day, most of us will need to print out at least some of our files on paper. In this chapter, we tell you how to do that. Using the methods described here, you can print text files, BASIC programs, and *data* files. You can print files you have entered and saved yourself or those created by a BASIC program. You can also print files you have received through the modem from another computer or from a cassette tape. You can even make a BASIC program give the commands to generate hard copy, if you don't want to do it yourself.

There are four ways to print a file: the **PRINT** key, the **CTRL** Y option, the **F4** "LPRINT" key, and the BASIC commands LCOPY, LPRINT, LLIST, and SAVE. Some of these methods apply to all types of files, some are appropriate for text files only, and some are used mainly for BASIC program files. After we show you how to hook up a typical printer, we'll describe each of these methods.

You should be aware that the Model 100 can display more characters than most other devices, including printers. If your files include special characters created by using the **GRPH** and **CODE** keys, you may find that these characters do not appear as you expected on the printed copy. In addition, graphic images drawn on the Model 100 screen by a BASIC program, using commands such as **LINE** and **PSET**, cannot be printed.

What You Need

To use the printing methods described here, you need three pieces of equipment: a Model 100, a cable to connect the Model 100 to the printer (Radio Shack part number 26-1409, shown in Figure 7-1), and a parallel printer. We're going to show you in detail how to hook up a popular dot-matrix printer, an Epson® MX™ or FX™, but the instructions are applicable to any parallel printer, whether it's a dot-matrix or letter-quality printer.

Hookup to a Printer

First, look at the printer cable. The smaller plug goes into the back of the Model 100 (see Figure 7-2). Find the port labeled **Printer** and gently plug in the cable. There's only one way the cable fits. If you're having difficulty, turn the cable over.

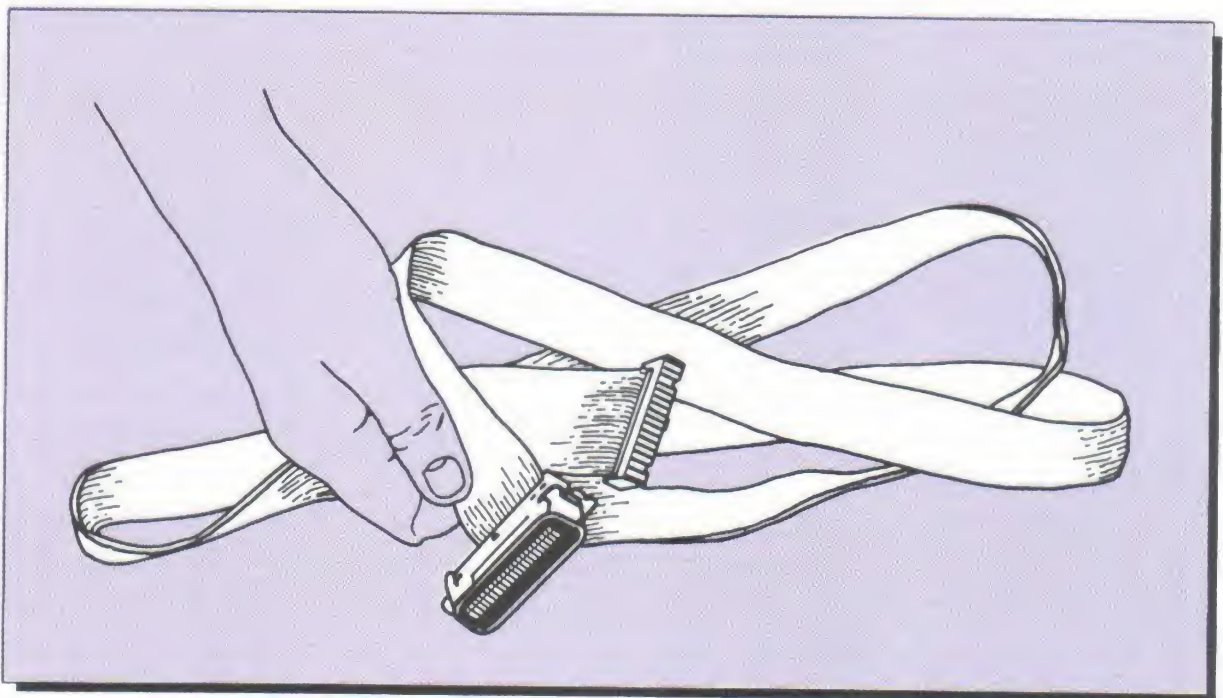


Figure 7-1. Radio Shack's cable connects the Model 100 to most printers

Now attach the larger end of the cable to your printer. This cable attachment also fits in one direction only. Watch what you're doing!

If you've been using your printer with a Radio Shack or Atari computer, you are ready to start printing as soon as your cable is hooked up. If, however, you've been using your printer with an Apple or IBM computer, there's one more step: giving automatic control of *line feeds* to your printer. These computers, via software, send line feeds to the printer every time they send a carriage return. Line feeds cause the printer to move the paper up one line at the end of each line. Radio Shack products and some other computers, on the other hand, do not automatically send line feeds to the printer after each carriage return.

Most dot-matrix printers, such as the IBM 80 CPS and the Epson MX-80, MX-100, and FX-80, have a switch for controlling line feeds. This is often a tiny switch called a DIP switch (see Figure 7-3), located in a fairly easily accessed location. On the Epson MX series, DIP switch number 2-3 controls the line feeds and should be on when you're using the Model 100; on the Epson FX printers, DIP switch number 2-4 controls line feeds and should be on when you're using the Model 100. If you use your printer with two different computers, you may have to flip this switch back and forth as you switch between your Model 100 and your IBM or Apple. It's a good idea to have a small screwdriver to flip the switch, unless you have long, skinny fingernails.

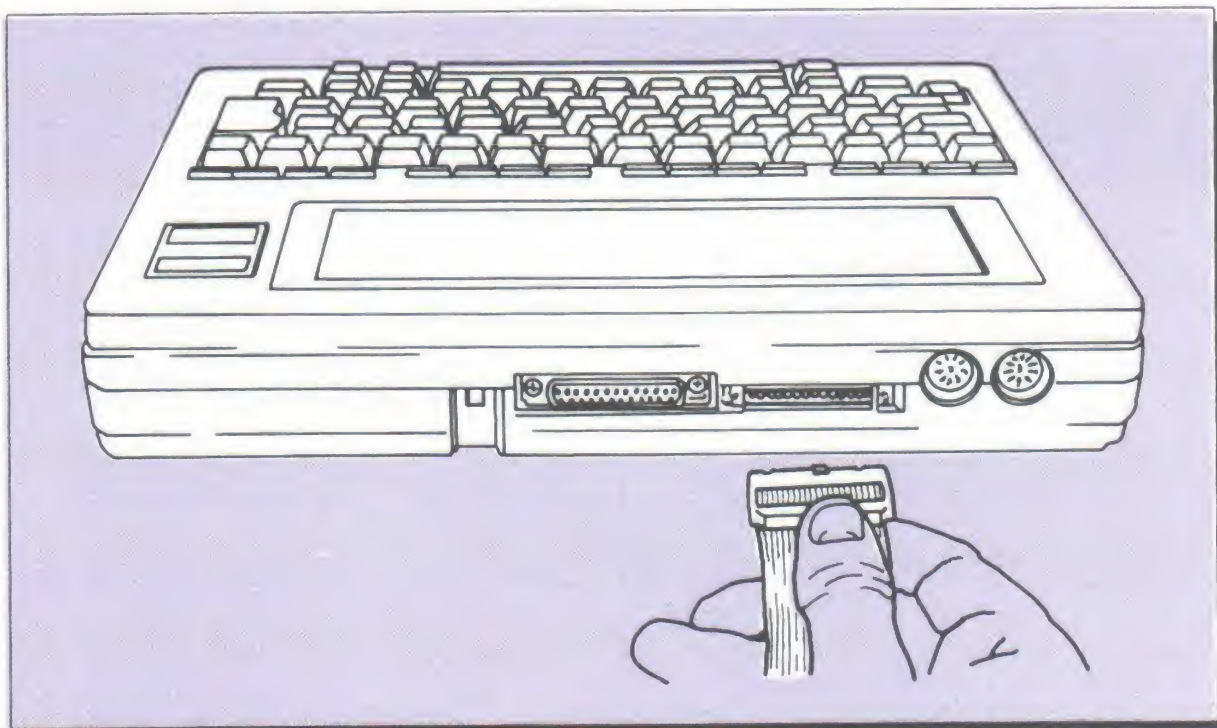


Figure 7-2. Plugging the printer cable into the Model 100

To find the exact location of this switch on your particular printer, you should consult the manual that comes with the printer. The settings for this and other switches are usually listed in the index under “DIP Switches” or “Function Switches.”

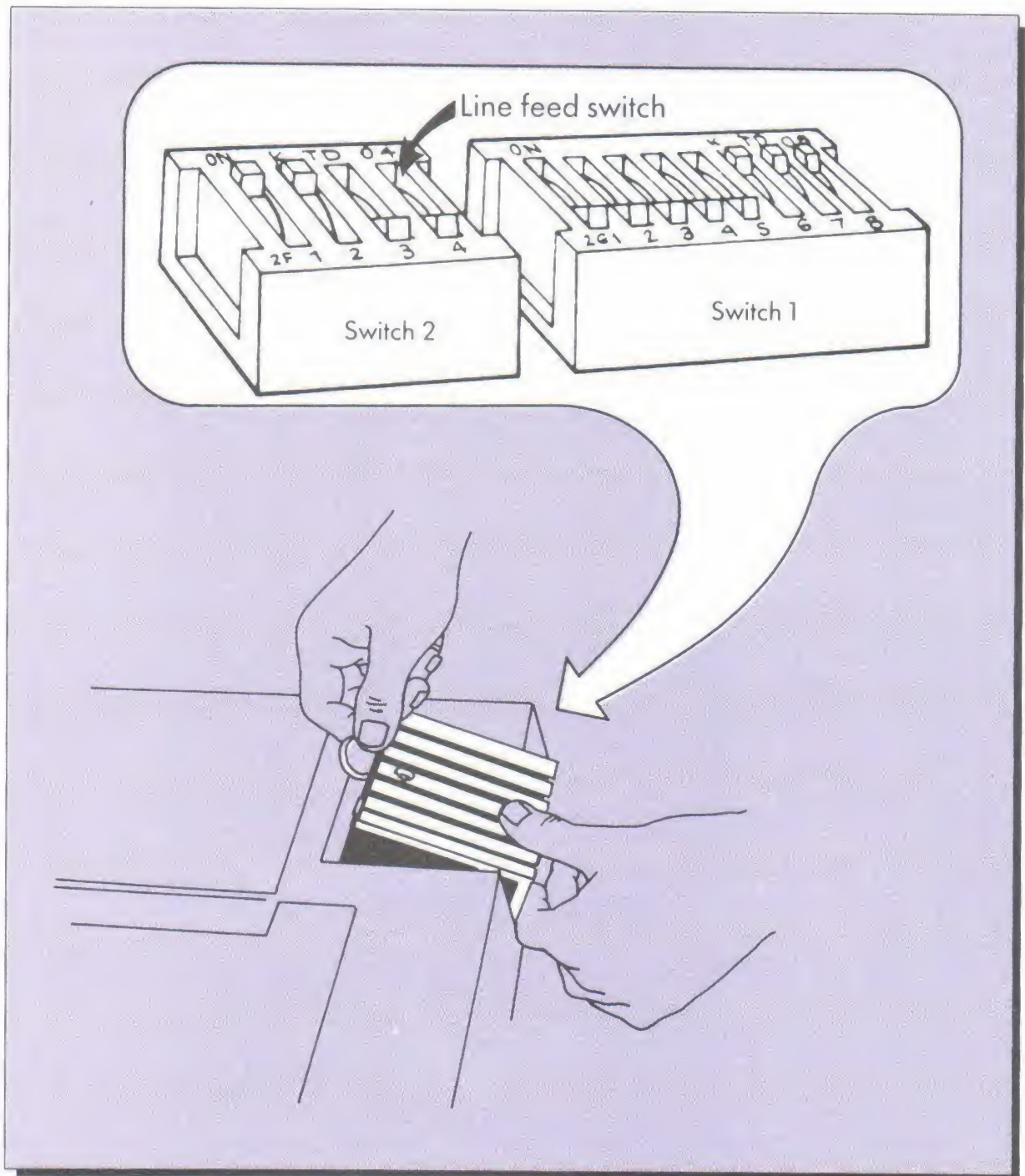


Figure 7-3. Dip switches on an Epson FX-80 printer

Printing the Screen Image: **PRINT** Key

Now that your printer is hooked up and “talking” to your computer, let’s look at some of the ways it can be put to use. The first and simplest method we’ll examine is printing the screen image.

Sometimes you’ll want to have a copy of exactly what’s on your Model 100 screen. For example, you might want to refer to the beginning of a letter that you’re typing. Instead of continually moving the cursor back to the beginning, you can just print it out and keep it handy for reference. Pressing the **PRINT** key causes the screen contents to be printed on the printer exactly as displayed.

Suppose you want to print out the first eight lines of the NOTE.DO file. You can enter the NOTE.DO file (by positioning the cursor over NOTE.DO on the main menu and pressing **ENTER**) and view the first eight lines of the file on the screen. Now press the **PRINT** key. The first eight lines of the file will be printed out exactly as they appear on the screen. (Remember, however, that special characters created by using the **GRPH** and **CODE** keys might not appear as expected on the printout.)

The **PRINT** key can also be used to print BASIC programs. If you wish to periodically print the screen image as you create a new BASIC program, simply press **PRINT** whenever you wish (after typing a line of code and pressing **ENTER**). (This has the same effect as typing the BASIC command LCOPY, which we’ll describe later in the chapter.)

Printing Whole Text Files: **CTRL** Y

You will often want to print out more lines at once than fit on the screen. In fact, you will usually want to print out an entire file. This might be a prose file you created with TEXT, or it might be a BASIC program. You could print a file, using the **PRINT** key technique described previously, by printing one screenful, scrolling the screen so the next eight lines show, printing another screen, and so on. But this technique is tedious and inefficient.

A better way to print an entire text file is to use the **CTRL** Y key combination, which will cause an entire file to be printed, provided that the file is “active”, or presently being edited. To use this technique, you simply press **CTRL** Y (that is, hold down the **CTRL** key while pressing Y). This causes the entire contents of the text file to be printed on the printer from beginning to end. Text will be printed at the width you specify; carriage returns in the text will start new lines on the printer.

For example, enter the NOTE.DO file to view the first eight lines on the screen. Now press **(CTRL) Y**. You will be prompted for entry of the printed line width. If the width shown in the prompt is what you want, simply press **(ENTER)**. Otherwise, type a new number, then press **(ENTER)**.

To print BASIC files with **(CTRL) Y**, you must be in EDIT Mode. Under EDIT Mode, BASIC files are the same as text files, and **(CTRL) Y** will cause the lines currently being edited to be printed out. (For more on the EDIT Mode, see Chapter 6.)

To print a BASIC file when you are not in EDIT Mode, you must use LLIST or SAVE "LPT:", as described next.

Printing BASIC Programs: LLIST

As we've mentioned, you can print a BASIC file using the **(CTRL) Y** key combination *if* you are in EDIT Mode. Most of the time, however, you'll find it more convenient to print a program listing from BASIC.

Using the command LLIST, you can print whole BASIC files, or you can select a range of lines to be printed. For example, you can enter any BASIC program by positioning the cursor over the word *BASIC* on the main menu and pressing **(ENTER)**. Press **(F2)** to load a BASIC program that you have previously typed in and saved. The screen will display the prompt "Load". Type the name of the BASIC program and press **(ENTER)**. For example:



When the program is loaded, the screen will display the message Ok. Now, to print the entire program, simply type LLIST and press **(ENTER)**.

To print only a portion of the program, type LLIST and give the range of line numbers you wish to print, then press **(ENTER)**. For example, typing LLIST 10-50 **(ENTER)** will cause only lines 10-50 to print out. With LLIST, you are not offered a choice of print width. Each line will be printed continuously across the paper until a carriage return signals the end of a line.

Printing BASIC Programs: SAVE "LPT:"

You can also print whole BASIC files using the SAVE command. To do this, enter BASIC and press **(F2)** to load any program you have previously typed in and saved. The screen will display the prompt "Load". Type the name of the BASIC program and press **(ENTER)**.

When the program is loaded, the screen will display the Ok prompt. To print the entire program, press **(F3)**. The screen will display the prompt "Save". Type "LPT:" and press **(ENTER)**:



As with LLIST, this method does not offer a choice of print width. Each line will be printed continuously across the paper until a carriage return signals the end of a line.

A BASIC Print Program: LPRINT

So far we've talked about ways that you can enter commands directly to make your computer print images from the screen or print entire files. However, it's also possible for a BASIC program to control the printer. If you're writing a BASIC program, you'll find many reasons to send the program output to the printer. Perhaps your program generates mailing labels or a list of mortgage payments. In either case, a hard-copy printout is more useful than having to read the names and numbers off the screen.

BASIC provides a command for sending output to the printer — LPRINT. This command will print strings or numerical constants, or entire files, but you need to know how to program in BASIC to use it. For example, the following lines will ask you to type a name and address. Then they will print this name and address on a mailing label or envelope.

```

10 REM ENVLOP.BA
20 PRINT"PRINT AN ENVELOPE OR MAILING LABEL
30 INPUT"WHAT IS THE NAME";N$
40 INPUT"WHAT IS THE COMPANY NAME";F$
50 INPUT"WHAT IS THE STREET ADDRESS";L$
60 INPUT"WHAT IS THE CITY";C$
70 INPUT"WHAT IS THE STATE";S$
80 INPUT"WHAT IS THE ZIP CODE";Z$
90 PRINT"READY PRINTER, PRESS ANY KEY"
95 A$=INKEY$:IF A$="" THEN 95
100 LPRINT N$
110 LPRINT F$
120 LPRINT L$
130 LPRINT C$;"", "S$;" "Z$
140 INPUT"ENTER 1-ANOTHER LABEL OR 9-EXIT";A
150 ON A GOTO 10,,,,,,,999
999 MENU

```

(For examples of programs that build and print a full mailing list file, see Chapter 5.)

Printing in SCHEDL and ADDRSS: (F4)

The SCHEDL and ADDRSS programs have built-in functions for printing selections from NOTE.DO and ADRS.DO, which means that you can obtain hard copy of your calendar of events, as well as all your important addresses and phone numbers.

For example, enter the SCHEDL program (by positioning the cursor over SCHEDL on the main menu and pressing (ENTER)). The screen will clear and display the prompt "Schd:". Press the (LABEL) key to view the F key functions under SCHEDL; then press (F5) for the LFND function. Now, to print the entire NOTE.DO file, simply press (ENTER). To print only those entries planned for the date 9/13, for example, type that date and press (ENTER):



All lines in the NOTE.DO file that include the date 9/13 will be printed out on the printer.

The **F5** key performs the same function for the ADDRSS program — printing lines from the ADRS.DO file.



8

Storing Files on Cassette

Concepts

- What you need to use a cassette
- How to hook up your Model 100 to a cassette
- Saving and loading text files
- Saving and loading BASIC files
- Saving a BASIC file in text format
- Running a BASIC program from cassette

*T*he Model 100 is so handy and useful that you'll probably soon notice the available amount of RAM memory — the "bottom line" on your screen — growing smaller and smaller. As we mentioned earlier, the Model 100 has a limited storage capacity.

Watch Your Bottom Line

Fortunately, there are various other storage options available besides RAM memory. You can transfer your files to another computer via the Model 100's RS-232 port (see Chapter 11) or via the telephone lines (Chapter 9). You can use the storage space available to you on CompuServe (Chapter 10), and you can store files on the disk-drive video interface, if you have purchased this extra option (Chapter 12). However, the simplest permanent storage device for the least cost is the cassette tape recorder, the subject of this chapter.

Cassette storage is very versatile, and in the following sections you'll see how easy it is to make the cassette a handy extension of your Model 100. Once you get the basic techniques down, you'll find you can easily run programs that are stored on cassette tape. You store important data and call it up at will.

What You Need

In order to save files to cassette, you need your Model 100, the Radio Shack CCR-81 Computer Cassette Recorder (#26-1208), and a high-quality computer cassette (like Radio Shack's #26-302). The cassette recorder includes a cable that connects it to the Model 100.

You can use other kinds of cassette recorders with the Model 100, but you will generally sacrifice something in the performance. With some recorders, you cannot rewind or fast forward without removing all the cables; with others, you may have to adjust the volume to full blast. The CCR-81 cassette recorder was designed for use with computers, whereas most other recorders were not. Computers place different demands on cassettes than normal audio use. The fidelity, so critical on stereo system cassette recorders, is secondary for the Model 100's purposes. More important is the ability both to start and stop at the proper time and to find the beginning and end of a file.

How to Hook Up Your Model 100 to a Cassette Recorder

The CCR-81 cassette recorder can be powered by four size C batteries, or it can be plugged in via the AC cord that comes with it. If your electrical power is erratic — does your TV flip when your clothes dryer changes cycles? — then you'd do well to use batteries. Electrical noise can interfere with the reading and writing operations of your recorder.

It's easy to hook up the Model 100 to the cassette recorder (see Figure 8-1). First, look at the cable that comes with the recorder; the cylindrical jack goes to the round port on the back of the Model 100 labeled Cassette. It can only be inserted in one direction. Now connect the black plug into the recorder jack labeled EAR. The large gray plug goes into the jack labeled AUX, and the small gray plug goes into the jack labeled REM. Now you're set to go!

Some Special Tips

Cassettes have been used for storage on microcomputers for a long time. Today, however, most full-size microcomputers rely on "floppy" diskettes or a "hard" disk for storage. Diskettes and hard disks offer advantages over cassette. It's easier to find something on a circular disk than on a linear tape. You cannot list a "directory" of all the files on a cassette tape, which you can do with a floppy diskette. Why, then, does the Model 100 give you

the option of cassette storage? One reason is expense. The equipment required to use floppy diskettes, namely the disk-drive video interface, is far more costly than a cassette recorder. Also, the cassette recorder is less bulky. Cassette storage has caused a lot of unnecessary frustration in the past. However, by following a few simple guidelines, you can use cassette storage as easily and reliably as any other storage system.

Power Supply

To avoid electrical noise, it is safest to use batteries to power both the Model 100 and the cassette recorder. The batteries should be as fresh as possible. It's a good idea to get an inexpensive battery tester (like Radio Shack's #22-031) to check your spare batteries from time to time and make sure they're really fresh. You can't always assume the batteries you buy are good — they may have sat on the shelf long enough to seriously degrade their performance.

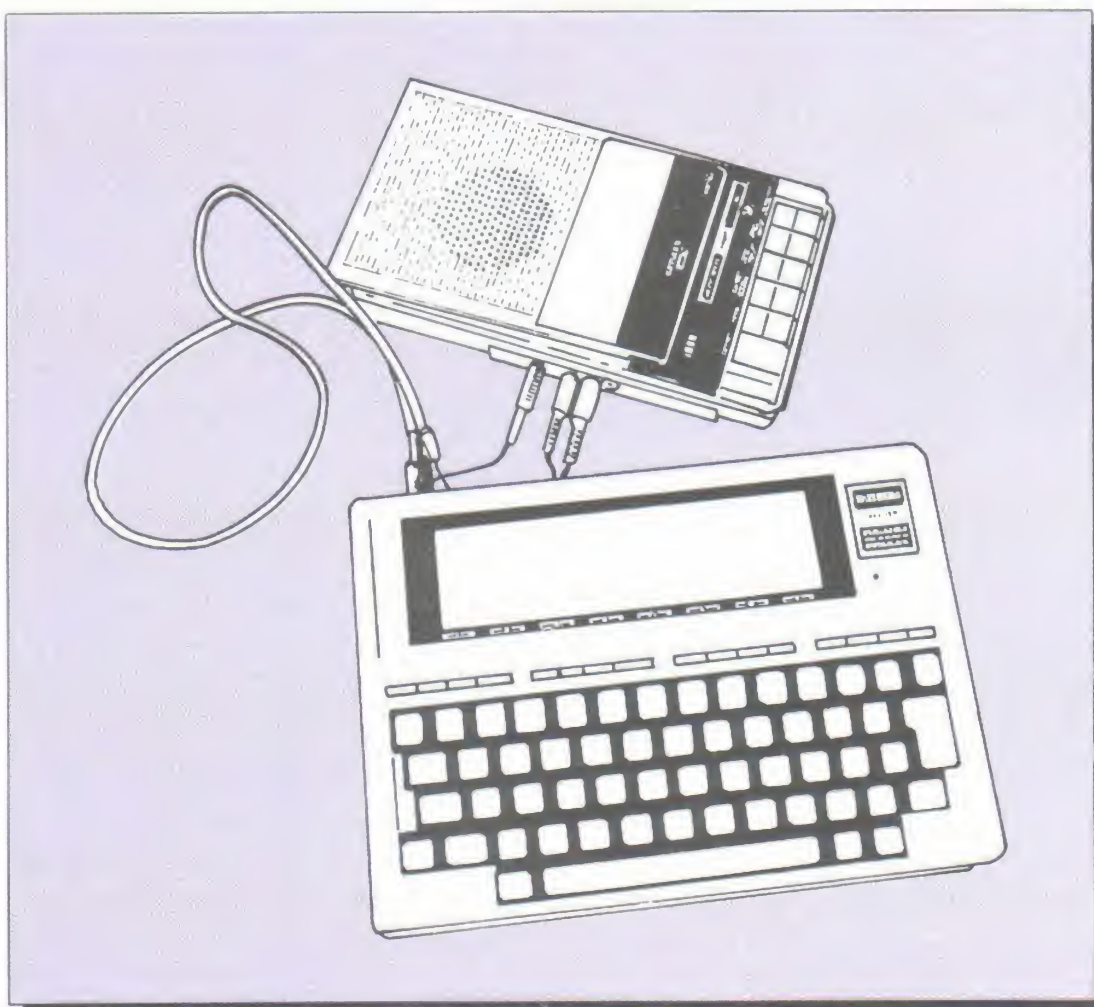


Figure 8-1. The Model 100 hooked up to the Radio Shack CCR-81 computer cassette recorder

Cassettes

Use a high-quality cassette tape. Spending a little extra will be more than worth it, if you don't want to lose files you may have spent hours building. By far the best are "leaderless" tapes made specifically for computer use.

You can visually check the quality of a cassette tape by inserting a pencil or pen in either tape reel center and rolling the tape. Look at the tape to be sure it has no ripples, creases, or discolorations. To maintain the quality of your cassettes, store them in the case at all times; they should never be stored in excessive heat or cold. If you keep a cassette in your recorder, be sure the tape head is retracted when not in use.

If you write down the number readings from your tape counter as you begin and end each file, you will be able to locate it more easily. Also, write down the filename on your cassette and record the name it had on the Model 100. By recording the Model 100 filename, such as PROG.BA or MEMO.DO, you'll automatically know if it's a text or BASIC file.

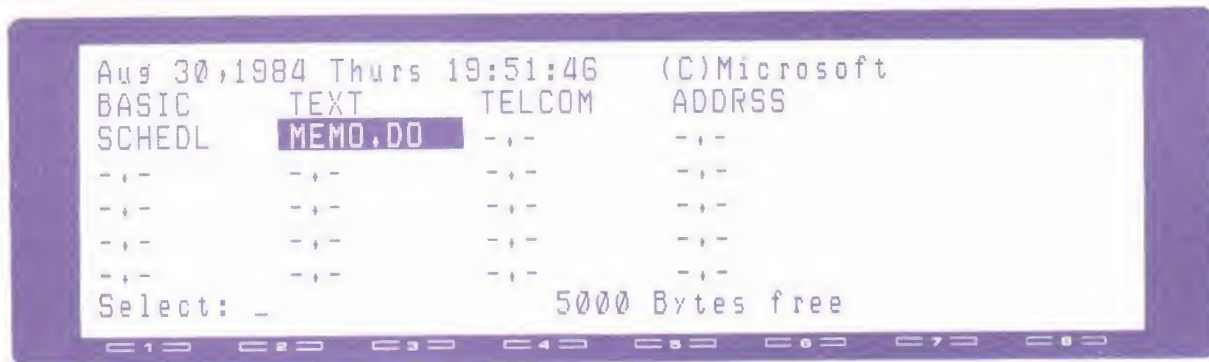
Saving and Loading Text Files

The procedure for transferring text files to and from a cassette recorder is slightly different for text than for BASIC files. We'll talk about text files first and then explain how to transfer BASIC files.

Saving a Text File to Cassette

Place the cassette in the recorder and set the counter to 000. If you are not using leaderless computer cassettes, it's important to move the *leader* — the tape at the beginning of the cassette — past the tape head *before* you start recording. You can't record on it. Once you've hooked up the Model 100 to the cassette recorder, you cannot simply press the PLAY button to get past the leader. The tape can only be advanced by using the FAST-FORWARD button. (You can also advance it with BASIC commands: the command MOTOR ON turns on the cassette recorder's advance mechanism; MOTOR OFF turns it off.) To get past the cassette leader, advance the tape until the counter reads "002".

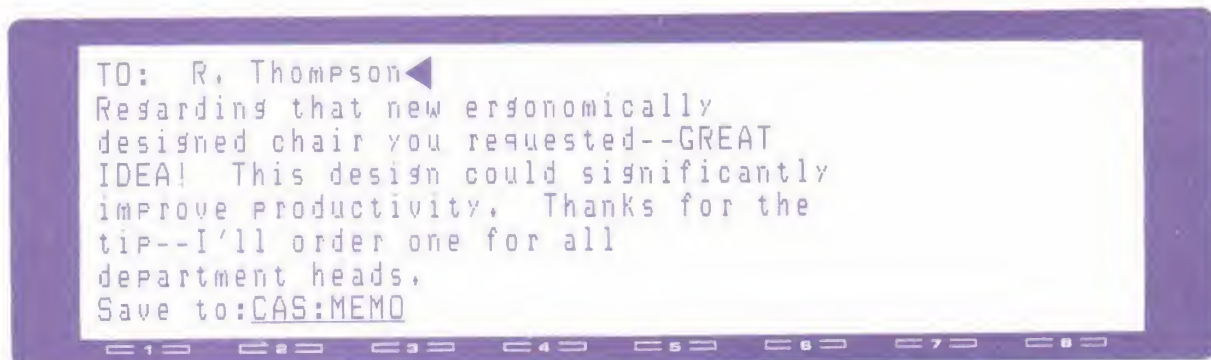
Now you're ready to save a text file from the Model 100 to cassette. Place the cursor on the name of the file you wish to save to tape and press **ENTER**.



In this example, we'll use our file from Chapter 3 called MEMO.DO. You can use any text file you've already stored on your Model 100.

Be sure to record the filename and counter reading at this point, using the card inside the cassette box. Most such cards have blank, ruled lines on the reverse side for this purpose. Now, simultaneously press the PLAY and RECORD buttons on the cassette recorder. The reels will not start turning yet, but the buttons remain depressed.

The screen should now display your text file. To begin the saving procedure, press **(F3)**. When the screen prompts "Save to:", enter CAS:, followed by the filename you want on the tape. The name can be up to six letters long and can be the same name the file has on your Model 100 (without the ".DO" or ".BA" extension).



As soon as you press **(ENTER)**, the cassette recorder will start turning, the counter will start counting, and your file will be saved to the cassette. The screen on your Model 100 does not change, even though the entire file, including the part you can't see, is being saved.

When the save is complete, the cursor will resume flashing on the screen. Also, the tape will stop turning on the recorder, and the counter will stop counting. Record this ending number on the counter so that you will know where to begin saving the next file.

You can then return to the main menu by pressing **(F8)**. If you want, you can now delete the file from memory by entering BASIC and entering KILL "MEMO.DO". That's all there is to it! Because you have a clear record of what's on each cassette, you'll always have easy access to anything you store on tape.

It's a good idea to leave a little bit of blank space between each file, so advance the tape a little before you save another file to disk. You can use the FAST-FORWARD button or enter BASIC and type MOTOR ON **(ENTER)** MOTOR OFF **(ENTER)**. Typing MOTOR ON advances the tape, but only until you type MOTOR OFF, usually about two ticks on the counter. Be sure to record the counter setting for the beginning of the new file. The more organized you are, the easier it will be to work with many different files.

Here's a summary of the steps involved in saving a text file to tape:

Saving a TEXT File

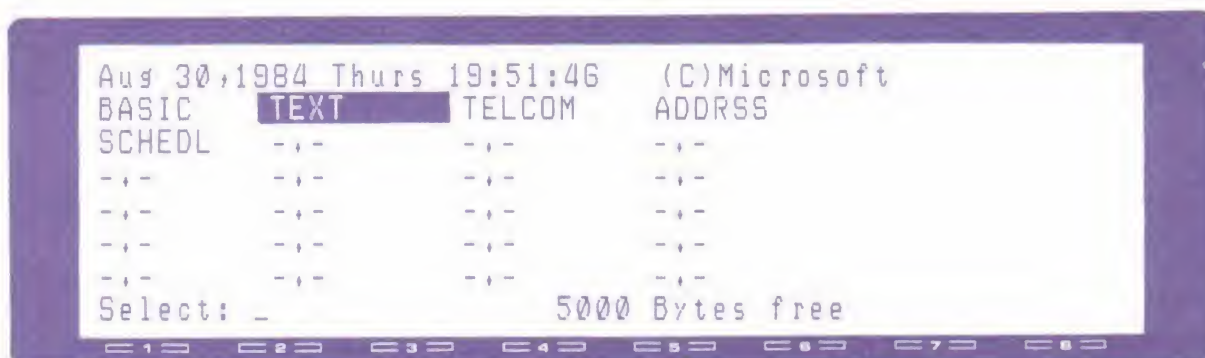
1. Position the tape where you want the file saved. Simultaneously press the PLAY and RECORD buttons on the cassette recorder.
2. Place the cursor over the name of the file you want to save and press **(ENTER)**.
3. Press **(F3)**, "Load", and enter CAS:, followed by the name of the file (first six letters only), then press **(ENTER)**.
4. The cassette recorder will start turning; when it stops, the file will have been saved.
5. Make a note of the number on the tape position counter, so you will know where to start when you save another file.

Loading a TEXT File

It's as easy to LOAD a text file as it was to save it. For our example, we'll be loading the text file MEMO from cassette into our Model 100. Because you have written down the readings from your tape counter as you began and ended each file, you can now use those readings to go directly to the file you want to load.

After you've hooked up your Model 100 to the cassette recorder, place the cassette in the recorder and set the counter to 000. Look up the counter position for the file you want and advance the tape to that location by pressing the FAST-FORWARD button.

To load the file, enter the TEXT program on the Model 100.



When you see the prompt "File to Edit?", you can enter a new filename or use the original one. Enter the name you want the file to have in your Model 100's memory. In this case, let's call the file MEMO2. The screen is blank when the file appears, because there's nothing in the file yet.

Before you start loading the file from cassette, be sure the recorder volume is turned up to the 5 setting. With the tape in position just before the file you're going to load, press (F2). When you see the prompt "Load from:", enter CAS:, followed by the name the file has on the cassette tape:



Be sure you use the same name that you originally gave the file on the tape. As you can see from this example, it may be different from the name you give to the file in the memory of the Model 100. After you type in the filename, press (ENTER).

The tape begins to move; you will then hear a high-pitched noise when the recorder finds the file. The bottom line on the screen will change to read:

Found: MEMO

The file is now being loaded from the cassette tape to the Model 100. When the LOAD operation is complete, the file will exist on both the tape and the Model 100. When the file is completely loaded, the tape will stop turning, the high-pitched noise will cease, and, as if by magic, the beginning of the text of the file will suddenly appear on the screen!

Go to the main menu by pressing **(F8)**, and you will see the file MEMO2.DO. You can now edit the file and again save it to tape. You could call up an existing file and insert the file you load off the cassette tape into that file. If you have certain lines or paragraphs that you find yourself typing in often, you can save them to cassette as a text file. They will be appended to the end of the file. Then, when you need that text in a document, you can load the file from cassette. Of course, when you save the new edited version of a particular text file to cassette, you'll have to save it under a different filename and in a different location than the previous version. You can see how it might get tricky if you do a lot of editing on a large file. The basic steps, however, are easy. Here's a summary:

LOADing a File from Cassette

1. Advance the tape to the position of the file you want to LOAD.
2. Place the volume knob on the cassette recorder to "5" and press the PLAY button.
3. Enter the TEXT program and type in the filename on the Model 100.
4. Press **(F2)**, "Load", and enter CAS:, followed by the cassette filename, and then press **(ENTER)** to begin the transfer.

Saving and Loading BASIC Files

It's just as easy to store your BASIC programs on cassette as it is to store text files. The steps are a little different, and you have the option of storing your BASIC files in a format that the TEXT program can use, namely, as

an ASCII file. The cassette recorder is operated the same as when SAVEing and LOADing TEXT files, but different commands are entered on the Model 100 keyboard.

Getting Started

Place the cassette in the recorder and set the counter to 000. Remember, it's important to move the tape leader past the tape head *before* you start recording. Advance the tape by using the FAST-FORWARD button or by advancing it with a BASIC command. To get past the cassette leader, advance the tape to 002 on the counter. Prepare the cassette recorder the same way you would for a text file. Simultaneously press the PLAY and RECORD buttons. Nothing will happen, except the buttons will lock into position.

Saving a BASIC File

Now you're ready to save a BASIC file from the Model 100 to cassette. Instead of placing the cursor over the name of the BASIC file you wish to save, which would cause the program to automatically RUN, place the cursor over the word *BASIC* and press **ENTER**. All cassette-related operations for BASIC programs are done from the BASIC application program. In this example, we'll use a file called PROGRM.BA. You can use any BASIC file on your Model 100.

Now you're in BASIC, but before you can SAVE the file to cassette, you must load it into the BASIC program. Do this by pressing **F2** and, when you see the prompt, entering the name of the program stored in the memory of your Model 100:



Now that your program has been loaded into BASIC, there are two possible commands to save the BASIC file to cassette tape. One is CSAVE "PROGRM

ENTER). You can also press the function key **F3** and, when you see the prompt "Save to:", type in CAS:PROGRM **ENTER**):



The recorder starts turning, the counter starts counting, and your file is being saved, or "written", to the cassette. As with text files, the screen on your Model 100 does not change, even though the entire file, including the part you can't see, is being saved. You'll know that the procedure is complete when the cursor resumes flashing, the tape stops turning, and the counter stops counting.

Don't forget to write down the information on this file, as you did with text files, including the beginning and ending count, the name of the file on the Model 100, and the name you will give the file on tape.

Now that the program is safely on cassette tape, you can, if you wish, delete the file from your Model 100 by typing in KILL "PROGRAM.BA" **ENTER**. Or, if you want to keep the program stored on tape *and* in your Model 100, simply return to the main menu by pressing **F8**.

Here's a summary of the steps necessary to SAVE a BASIC file to cassette tape:

Saving a BASIC File to Cassette

1. Position the tape where you want the file saved. Press the PLAY and RECORD buttons on the cassette recorder.
2. Enter the BASIC program on the Model 100.
3. Press **F2**, "Load", and enter the filename.
4. Press **F3**, "Save", and enter CAS: followed by the name you want the file to have on cassette.
5. Press **ENTER**, and the cassette recorder will start turning; when it stops, the file has been saved. Make a note of the tape's position number on the counter.

Saving a BASIC File in TEXT Format

You can save your BASIC file to cassette in a format that can be used by the TEXT program. Follow all the above procedures except, when you type in the command to SAVE the file, add A. You can type CSAVE "PROGRM",A, or you can press (F3) and type CAS:, followed by the program name (in quotes), a comma, and A.

Save to "CAS:PROGRM",A

LOADing a BASIC File from Cassette

After you've hooked up your Model 100 to the cassette recorder, place the cassette in the recorder and set the counter to 000. Note the counter position for the file you want and advance the tape to that location by pressing the FAST-FORWARD button.

Get into BASIC from the main menu. You may now enter CLOAD:PROGRAM or use the function key (F2) and then enter "CAS, followed by the filename. Be sure you use the name of the file on the tape; it may be different from the name you give the file on the Model 100.



As the tape begins to move, you will hear a high-pitched noise. When the recorder finds the file, the bottom line on the screen will change to read.

Found: PROGRM

The file is now being loaded from the cassette tape to the Model 100. When the file is completely loaded, the tape will stop turning, the high-pitched noise will cease, and you will be in BASIC, ready to RUN or SAVE the program. When the LOAD operation is complete, the file will exist on both the tape and the Model 100.

Here's a quick summary of the steps you must take to LOAD a BASIC file.

Loading a BASIC File from Cassette

1. Advance the tape to the position of the file you want to LOAD.
2. Place the volume knob on the cassette recorder to "5" and press the PLAY button.
3. Get into BASIC on the Model 100; then press **(F2)**, "Load", and enter CAS:, followed by the cassette filename, and then press **(ENTER)** to begin the transfer.

LOADing and RUNning a BASIC File

You can LOAD your BASIC file from the cassette recorder and then RUN it, all in one command. Follow all the steps we discussed in the previous section except, when you type in the command to LOAD or CLOAD the file, add the letter R. That line from our example in the previous section would now read

```
Load from: "CAS:PROGRM",R
```

If you enter the command in this way, your program will RUN the moment it is completely LOADED into the Model 100. Don't forget to SAVE the file before you return to the main menu if you want to keep it in the Model 100's memory.

Another way to RUN your program immediately after it is LOADED is to follow all the above steps, except that after you enter BASIC, type in the following command:

```
RUN "CAS:PROGRM"
```

As soon as the file is LOADED, the program will begin RUNning.

Summary

Now you know how to use the cassette recorder to store BASIC and text files. In later chapters, we'll show you how to save files via RS-232, via modem, and to the disk drive and video interface.



9

TELCOM and the Marvelous Modem

Concepts

What you need

Modem hookup direct to phone line

Modem hookup through acoustic coupler

Sending and receiving files over the telephone line

*T*he world is getting smaller. In the old days, a hundred miles was far away, and to say hello to a friend in another state would take days or even weeks, depending on the weather and the number of obstacles the Pony Express might meet along the way. While early telephones were a vast improvement, it took a while to get a connection, and you were lucky if you could hear half of what was said. Eventually, of course, everyone got a phone and, as the phone company got bigger and bigger, service got better and better. You could talk to anyone almost anywhere in the world. A hundred miles became just a finger's walk away.

The New World

Today, voices are not the only things buzzing over the phone lines. Through the new world of telecommunications, one computer can "talk" to other computers all over the world. By sending information over the phone lines, businesses miles apart can be as mutually accessible as neighbors; students can have the information of great libraries at their fingertips; and children of all ages can match their wits with sophisticated computer wizardry by playing games of every kind imaginable, even though the game program may be running on a computer in a distant city.

All this has been made possible by a rather unassuming little box called a *modem*. Modem stands for "MOdulator-DEModulator", which tells the engineers among us something about the way the box works. Basically, a modem transforms the signals that computers send (digital) to signals that travel over telephone lines (analog) and back again.

The Model 100 hurls you into the center of this exciting new world, because a modem has been built right into the computer. By using the built-in TELCOM program, you, too, have access to virtually any computer in the world.

In this chapter, we'll show you how to hook up the Model 100 to the telephone lines, and, in the following chapter, we'll look at the different information services you can dial up with TELCOM and the marvelous modem. (If you already know how to hook up your Model 100 to the phone lines, you may want to skip this chapter and read on to Chapter 10 for a look at CompuServe® and other information networks.)

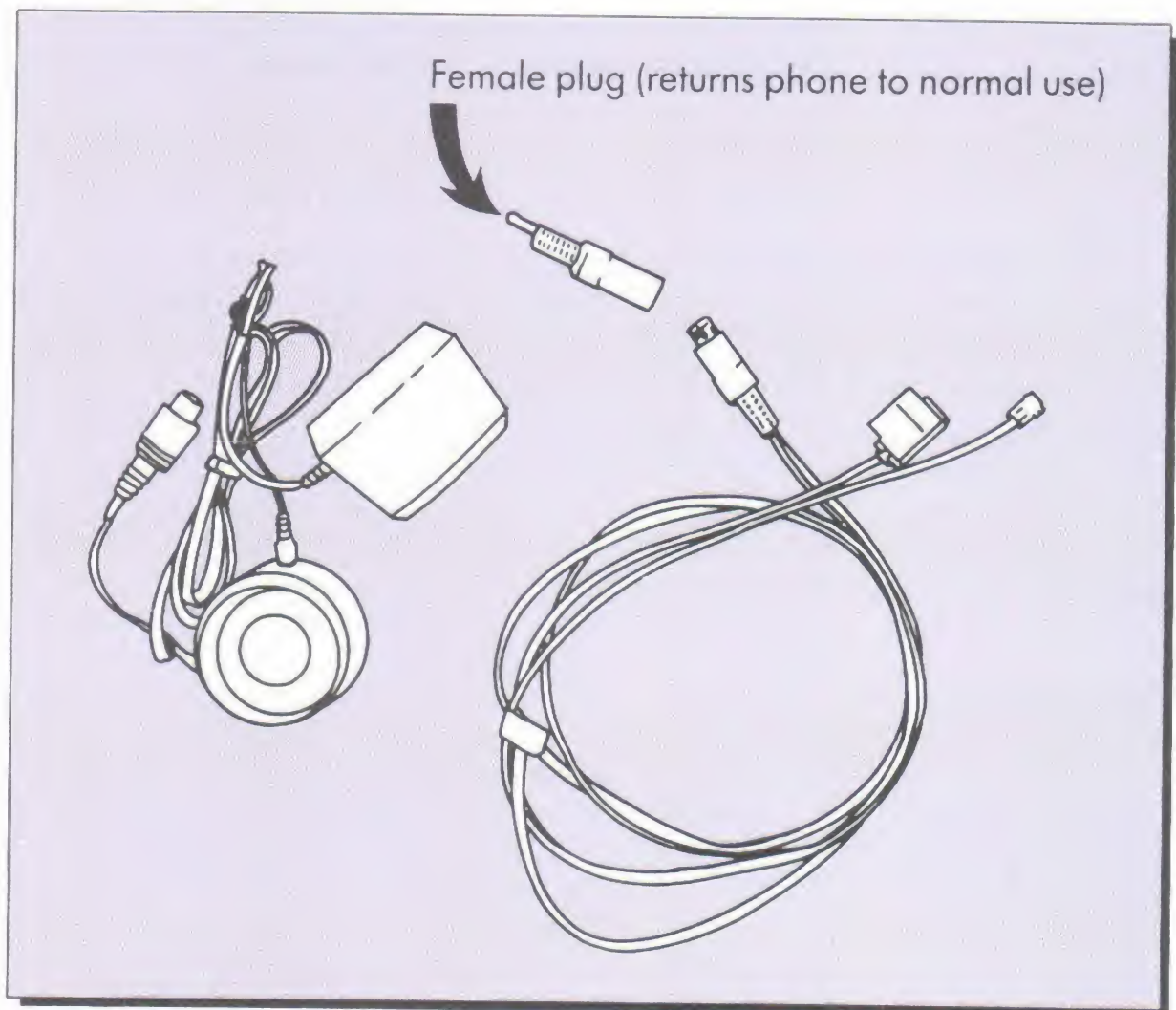


Figure 9-1. The acoustic coupler and the direct modem cable

What You Need

All you need to begin is your Model 100, a touch-tone telephone, and a modem cable — either the Model 100 Modem Cable (RS #26-1410) or the Model 100 Acoustic Coupler (RS #26-3805), as shown in Figure 9-1.

It's generally better to use the direct connection, if possible. With a direct connection, you'll get clearer communications than with the acoustic connection, which is prone to interference from outside vibrations and noise. And you *must* have the direct connection to use the autodial feature. However, a direct connection does require that your telephone be equipped with the modular telephone jack (see Figure 9-2). If you frequently find yourself in remote locations or in hotels without the modular jack, then you'll need the acoustic coupler. It will allow you to use *any* telephone receiver to communicate to the outside world over the telephone lines. However, if you know that you'll always be using a modular jack, the direct connection is the preferred approach.

Modem HookUp Direct to Phone Line (DIR)

Before dialing up NASA, CompuServe, or your neighbor's TRS-80 microcomputer, you have to connect the Model 100 to the telephone lines.

The Model 100 Direct Modem Cable (RS #26-1410) comes with a cable *and* the user's guides for two of the biggest, best information services avail-

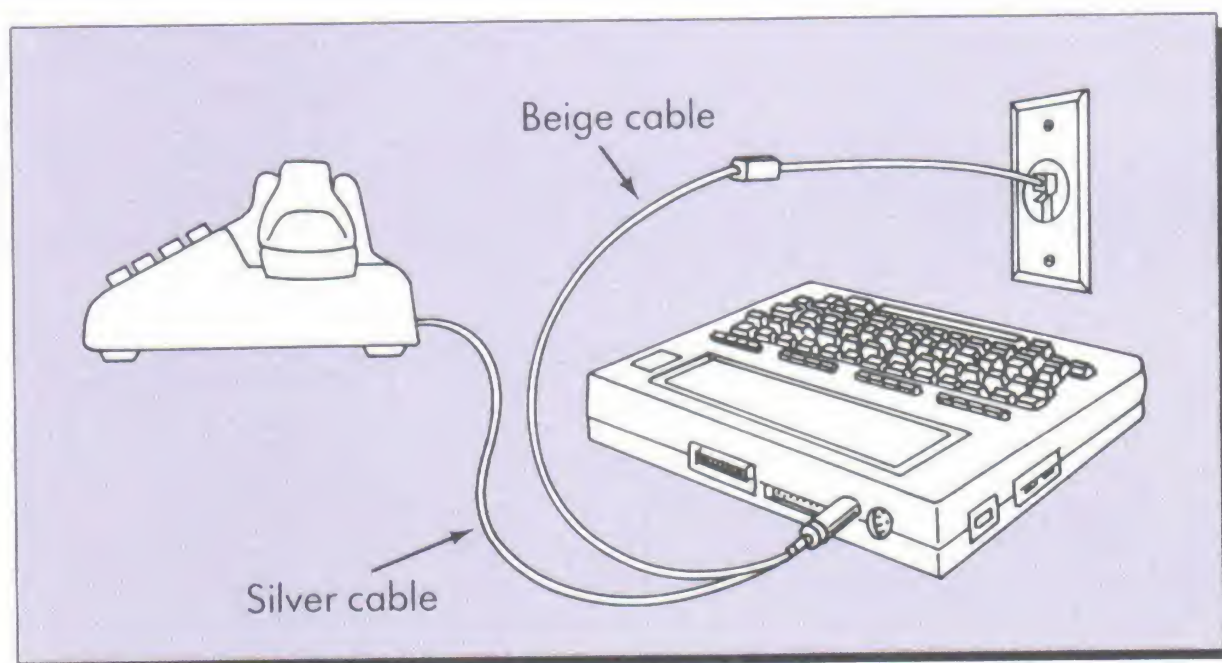


Figure 9-2. The modular telephone jack and wall outlet

able: CompuServe and Dow Jones News/Retrieval®. These guides also include one hour of free connect time on each service. We'll discuss these services in depth in Chapter 10. For now, let's concern ourselves with getting hooked up and learning the basics of TELCOM.

Look at the modem cable. The cylindrical jack goes to the round port labeled PHONE on the back of the Model 100; it can only be inserted in one direction. Next, remove the jack that connects the phone line into the back of your telephone and insert the gray line of the modem cable into that outlet. Now, take the line you removed from the back of the telephone and insert it into the connector on the end of the beige line of the cable. Or, remove the connector and plug the beige line directly into the wall outlet.

The "female" cylindrical plug is used when you have the modem cable connected to the phone lines and want to dial out on the telephone receiver instead of through the Model 100. You'll notice that if you try to dial out with your Model 100 hooked up, you'll get a dial tone and everything will be fine. If, however, you want to unplug the Model 100 from the cables and still use your telephone normally without unplugging all the cables, then you must attach the female connector to the cylindrical plug you removed from the Model 100. This is the only way you'll get a dial tone when the modem cable is still connected to the phone lines.

There are two switches on the side of your Model 100 marked DIR-ACP and ANS-ORIG. Referring to Figure 9-3, locate these switches. When you are hooking up by the direct connect method, be sure the DIR-ACP switch is set to DIR. As you've probably figured out by now, that stands for "DIRect connection".

One way to check if you've hooked up properly is to pick up the telephone receiver; if you hear a dial tone, then you are probably set to go.

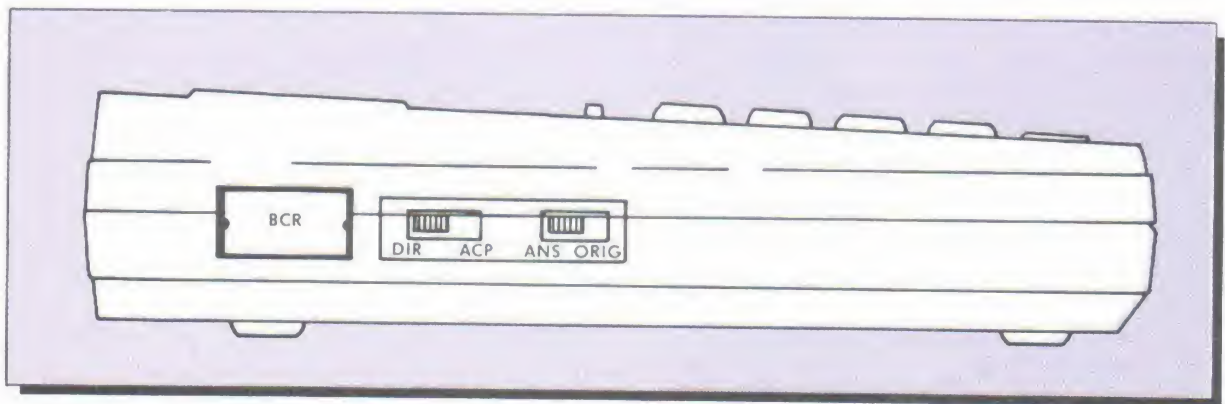


Figure 9-3. The DIR-ACP and ANS-ORIG switches

Connecting the Model 100 to the Phone Lines

- Step 1: Plug cylindrical jack into Model 100.
- Step 2: Plug silver wire into back of phone.
- Step 3: Plug line from phone into connector on beige wire.
- Step 4: Set the DIR-ACP switch to DIR.

In the next section, we'll show how to hook up the acoustic coupler. Feel free to skip it if you're using the direct connect cable only.

Modem Hookup Through Acoustic Coupler



As mentioned earlier, we cannot always use the direct connect method to access the telephone lines. Undoubtedly, the Miwok Motel in Outer Oshkosh is still using the telephones originally installed when the place was built. The lines are wired right into the wall and cannot be removed; they have no modular plugs. The acoustic coupler allows us to use any phone, regardless of the types of jacks it may use, and it connects to any standard telephone receiver (see Figure 9-4).

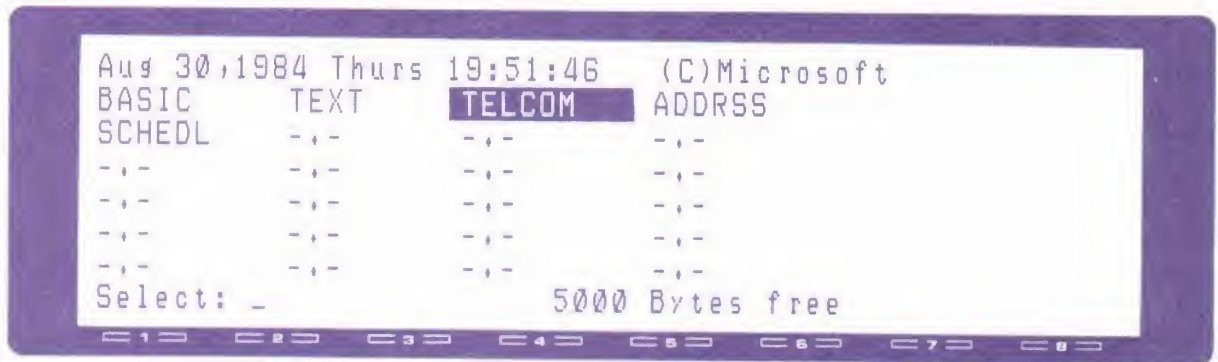
First, look at the acoustic coupler; connect the cylindrical end of the cable to the round port labeled Phone on the back of the Model 100. Next, insert the phono-size jack into the cup labeled Mouthpiece, then place this cup over the telephone mouthpiece so that it fits snugly. The other cup, labeled Earpiece, should fit snugly over the telephone earpiece. Once you have made this connection, place the handset on a flat surface. Finally, be sure to set the DIR-ACP switch to ACP (which stands for "Acoustic CouPler").

Acoustic Coupler Hookup

- Step 1: Plug cylindrical jack into Model 100.
- Step 2: Slip the cup labeled Mouthpiece over the mouthpiece of the telephone.
- Step 3: Slip the cup labeled Earpiece over the telephone earpiece.
- Step 4: Flip the DIR-ACP switch to ACP.

Getting Started

Once you've hooked up the the phone line, using either the direct connect or the acoustic coupler method, you are ready to run the TELCOM program. Use the  key to position the cursor over TELCOM and press .



Be sure that the ANS-ORIG switch is set to ORIG and that the DIR-ACP switch is set according to the cable you're using to connect to the telephone lines, (DIR for direct connect, ACP for acoustic coupler).

You are now in the *Entry Mode* of the TELCOM program. Unlike the other programs in the Model 100, TELCOM actually has two different modes of operation: the *Entry Mode* and the *Terminal Mode*. The function keys have different definitions in each mode. We use the Entry Mode to dial our connection to another computer and the Terminal Mode to do the actual "talking" to the other computer.

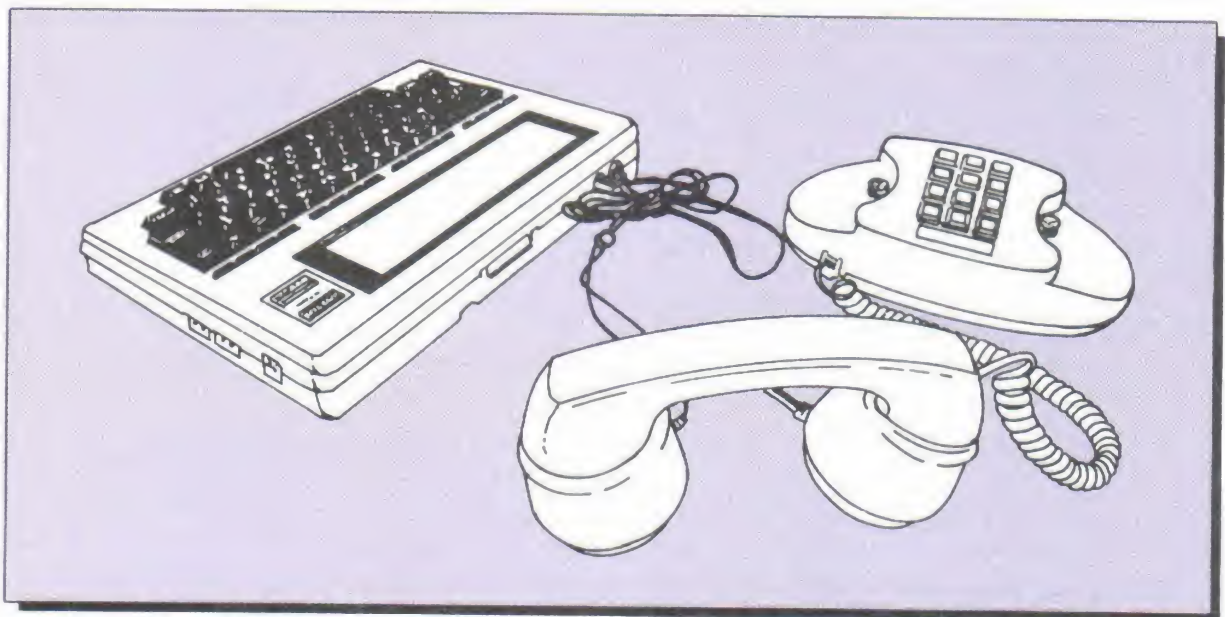
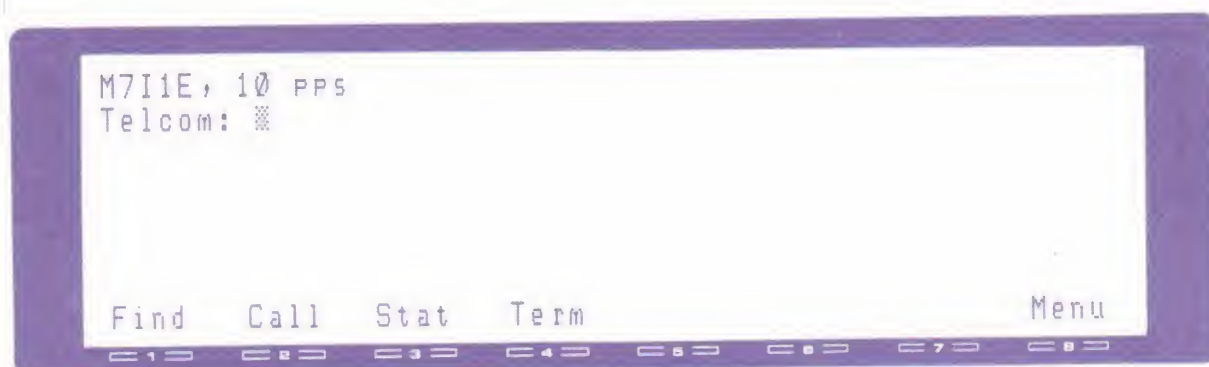


Figure 9-4. The acoustic coupler hookup



Communication Parameters

Before you can communicate with another computer, you must be sure that both your computer and the one on the other end are speaking the same kind of language. In this case, though, the language is in “data” words. Both computers must be operating at the same speed, using the same data word length, and using the same error-checking procedures. If that sounds complicated, don’t feel intimidated. It’s not necessary to be an engineer to use the Model 100’s TELCOM program. The standard or “default” settings are the ones you’ll use most of the time. It is a good idea, however, to have some general understanding of the possible different settings.

Look at the upper left-hand corner of the TELCOM screen. The first line, M7I1E,10 pps, tells you what the current settings are. These settings are called the *communication parameters*. Translated into English, they mean the following:

M “I am enabling the Model 100 modem. The speed of the transmission will be 300 bits per second (300 baud)”.

7 “Each character has seven bits of basic data (data word)”.

I “I am ignoring parity for error checking”.

1 “Each character will have one bit to signal the end of that character”.

E “Yes ma’am, I’m going to send in XON/XOFF protocol”.

Don’t worry about what all this means, but, if you can, check with the operator of the system you’ll be sending to and make sure his or her communication parameters are the same.

CompuServe® uses the setting M7I1E; Dow Jones News/Retrieval®, on the other hand, turns off XON/XOFF protocol with a setting of M7I1D.

If you need to make any changes, use the STAT function. Press **F3** and the word *Stat* appears after the *TELCOM* prompt. Then enter the settings you need in the same format and in the same order as the default settings shown above. You should always have a total of five characters

(letters or numbers). It doesn't matter if the letter entries are in upper- or lowercase; TELCOM will translate them to uppercase for you. Refer to the illustration below to see what all the possible settings are. After you enter the new settings, press **ENTER**.

Possible Settings for TELCOM Program

To set this:	To this:	Enter this:
Transmission Speed	Modem (300)	M
	75 baud	1
	110 baud	2
	300 baud (RS-232 only)	3
	600 baud	4
	1200 baud	5
	2400 baud	6
	4800 baud	7
	9600 baud	8
	19200 baud	9
Word Length	6 bits	6
	7 bits	7
	8 bits	8
Error checking	Even parity	E
	Odd parity	O
	No parity	N
	Ignore parity	I
XON/XOFF Protocol	Enable	E
	Disable	D
Pulse Rate	10 pps	10
	20 pps	20

Once you've changed the settings, the line in the upper left-hand corner will not have changed. To verify that the current settings you've entered are correct, type **F3** **ENTER**; the new status settings will appear below the prompt line.

Getting Online the First Time

Once you have set the proper communication parameters, you're ready to dial the other computer. In our example, let's say you've typed an important memo while sitting on Waikiki Beach. Now, in your hotel room, you're ready to send the memo back to the home office. The Model 100 is connected to the phone line with the direct cable. You're in the TELCOM program, and the default settings are the proper communication parameters for communicating with your home office computer. Ready to dial? Press **(F2)** and the word *Call* appears on the screen. Now you can type in the number you want to call. Let's say the number is (212) 987-6543.



As the number is being dialed, you may find you need to pick up the receiver. If you are using an acoustic coupler, hold down the dial-tone button and *then* press **(F2)**. Release the dial tone button at some point during the dialing.

Dialing from ADRS.DO

It's a good idea to enter the telephone numbers of other computers in your address file ADRS.DO. You can use the FIND function to bring up a number if it has already been entered in your ADRS.DO file. Instead of pressing **(F2)**, as in the above example, press **(F1)**. The word *Find* appears on the screen. Now you can type in the name you want to call.



As in the ADDRSS program, you needn't type the whole name in order for the Model 100 to find it.

Notice that the function key definitions have changed slightly. If the number that appears on the screen is not the one you want, you can use **(F3)**, "More", to search for the next occurrence in your ADRS.DO file of the name you typed in. If you realize that the number is not in ADRS.DO, you can press **(F4)**, "Quit", to return to the first TELCOM menu, where you can press **(F2)**, "Call", and type the number to be called manually.

This is an exciting moment! The screen says Calling MICHAEL COBB:, and the numbers begin to appear as the computer dials them as if by magic.

As soon as the dialing is completed by either method, you must go into Terminal Mode.

The End of the Line — Terminal Mode

When you reach the answering party — the computer you're calling — you must get into the *Terminal* Mode to communicate with it. The easiest way to know the right moment to enter Terminal Mode is to stay on the line while the Model 100 is dialing, listening for an open line after dialing is completed. What does an open line sound like? There will be a faint click, followed by a slight static noise. You'll learn quickly which moment is the right one. At that point, enter Terminal Mode by pressing **(F4)**.

The bottom line of the display now changes as the terminal shows the screen in Terminal Mode.



Note that Echo does not show up when you first enter TELCOM; it appears only when you use it. (Echo means "send to the printer" — more on that in a moment.)

When you have established a connection with the friendly machine in your home office, you can type in a message directly or send a file containing a memo to Mr. Cobb. Because Cobb himself probably would not be at the terminal on the other end, you can first say "Hi" to Rosie, the terminal

operator. You can talk to a terminal operator (sometimes called a SYSOP, for “System Operator”) simply by typing on the keyboard (in Terminal Mode only). This is called “chatting”.

A message you can send may look like this:

```
Hi Rosie!  
I have composed a message for you know  
who, I'm dropping the big one. Stay  
tuned. Wish you were here.
```

Robert

As you type in your message, it will appear simultaneously on the screen at the other end. The terminal operator then has the opinion of replying. Rosie may type back:

```
Hi!  
You lucky so-and-so, Am ready for you  
to send memo file, Bombs away!
```

Sending (Uploading) a File

Once you have established your connection using the terminal mode, you can send a file to the system on the other end. First, you must *upload* it to the host computer. Press **(F3)**, and the screen will prompt “File to upload?” Then type in the filename and press **(ENTER)**. The example used below is called NOTICE.DO.



The screen prompt “Width:” will appear. Width refers to the number of characters the Model 100 will transmit before it sends a carriage return (indicating the end of the line). You may enter any number between 10 and 132. Choose the width your home office’s word-processing program can accommodate. In this example, 65 is used.

```
Hi!
You lucky so-and-so, Am ready for you
to send memo file, Bombs away!

File to upload? NOTICE.D0
Width: 65

Prev   Down   Up    Full   Echo                               Bye
┌1┐   ┌2┐   ┌3┐   ┌4┐   ┌5┐   ┌6┐   ┌7┐   ┌8┐
```

As the file is being transmitted, it scrolls across the screen. It is also going across the screen of the host computer. Your screen may look like this.

```
Dear Mr. Cobb:

This is to inform you that my position
in your firm is hereby made vacant, I
am going to work for a company here in
Hawaii that designs ergonomic furniture
for computer operators. Thank you for

Prev   Down   UP    Full   Echo                               Bye
┌1┐   ┌2┐   ┌3┐   ┌4┐   ┌5┐   ┌6┐   ┌7┐   ┌8┐
```

That's all there is to it. Note that only text files may be transmitted or received over the phone lines. This means that if you want to send a BASIC program listing over the line, you must first convert it to ASCII, or text, format, as explained in Chapter 6.

Receiving (Downloading) a File

If you want to receive a file from the home office, the procedure is similar, except that you're now *downloading* the file from another computer. To do this, simply press **(F2)**. The screen asks "File to Download?" Enter the name you want the file to have on your Model 100, not necessarily the name of the file on the host computer. You are now ready for the host computer to send a file. As when a file is uploaded, the text appears across the screen as it is being sent. When you have received the end of the file (you'll hear a beep), stop downloading by pressing **(F2)** again.

When you are sending or receiving files, you are viewing eight lines of text at a time. In order to see the previous eight lines, press **(F1)**. Even though the Model 100 is still sending or receiving text, the screen scrolls up to the previous lines and then continues.

Function key **(F4)** indicates the *Duplex* Mode, which alternates between “full” and “half”. *Full* duplex, the default, means that the characters you type into the keyboard (remember our message to Rosie?) are sent directly to the host computer *and then* sent back to your terminal, where you see them. In *half* duplex, the characters are simultaneously sent to both your terminal and the host computer as you type them in. The mode you choose will probably depend on what mode your host computer likes. You get from full to half duplex by pressing **(F4)**; to get back to full duplex, press **(F4)** again.

Function key **(F5)** has a misleading label. *Echo* means that you can “echo” what’s on the screen to the printer. (See Chapter 7 to make sure that your printer is turned on and properly connected.) Press **(F5)** and the characters you see on the screen will also be printed out. You can use the *ECHO* function to get a hard copy of anything on the screen, whether you’re typing it in or sending or receiving information.

Saying Goodbye

Well, now that you’ve sent our memo to Mr. Cobb, you can hang up. In order to disconnect, press **(F8)**. The screen prompts “Disconnect?”. If you are ready to hang up, enter Y. That’s all you need to do; your session on TELCOM is over.

Here is a summary of the function key definitions for the Terminal Mode of TELCOM. You can see that the program is simple to run but that it also gives you a great deal of flexibility. In the next chapter, we’ll explore another exciting aspect of telecommunications and TELCOM: the commercial information services, CompuServe and Dow Jones.

TELCOM Terminal Mode Function Key Definitions

Key	Label	Function
(F1)	Prev	Displays the previous eight lines of text
(F2)	Down	Downloads a text file <i>from</i> another computer
(F3)	Up	Uploads a text file <i>to</i> another computer
(F4)	Full	Switches from full to half duplex
(F5)	Echo	Sends text on the screen to the printer
(F8)	Bye	Disconnects the Model 100 from the telephone line



Window on the World — Information Networks

Concepts

- Information networks are . . .
- What you need
- Logging on to CompuServe
- CompuServe's main menu and commands
- Movie reviews on CompuServe
- Electronic bulletin boards
- Dow Jones News/Retrieval service
- Coding an auto-log-on procedure

In the previous chapter, you learned how to use the Model 100's TEL-COM program to communicate with another computer over the telephone lines. In this chapter, we'll explore some of the information available in the vast, ever-growing world of *online databases*.

What exactly is an online database? Physically, it consists of a large computer with enormous storage capacity and a great many modems, so it can communicate with many other computers — such as yours — at the same time. This computer uses its storage capacity to hold information of many kinds. You can access this information by dialing up the large computer on the telephone and then using your computer to read whatever data you find interesting. (Online means that the large computer can connect to you over the phone lines; database means that it holds a lot of information.)

Information Networks

Through commercial online databases, you can now use the Model 100 as your window to a world never seen before. You can dial up information on news, weather, and sports from major newspapers around the world. Financial information of every kind is available at your fingertips. It's no longer necessary to call the airlines for flight information — the Official Airline Guide is online with current information about all the airlines' comings and goings. You can download software programs for free. You can match your wits against computer chess players and stretch your imagination in computerized fantasy adventures. The list of possibilities is endless. Every day, thousands of people are working at making more information available and easier to access.

For their part, Radio Shack started making it easy when they decided what to package with their modem cable. Every modem cable comes with a user's guide and one *free* hour of connect time to two of the largest and best information networks in the country — CompuServe Information Service (CIS) and the Dow Jones News/Retrieval (DJNR) service.

In this chapter, we show you how to use these services. Soon you'll be online, looking through the window of your Model 100 to an almost limitless world of information, communication, knowledge, and fun!

What You Need

All you need to get started is your Model 100, a touch-tone telephone, and the optional Model 100 Modem Cable (RS #26-1410), as shown in Figure 10-1. It is possible to use the acoustic coupler to dial up the information services, but it does not come with a user's guide for the two services; further, you cannot use an "auto-log-on" procedure (see the section later in this chapter) with the acoustic coupler. If you have the acoustic coupler and want free connect time to CompuServe and Dow Jones News/Retrieval, buy the direct modem cable anyway, as it is still much less expensive than paying the normal hookup fees charged by each service.

In this chapter, we'll be talking about using the direct connect modem cable only, as well as the user's guide and free connect time that come with it.

To dial up the information networks, you must always have the ANS-ORIG switch set to ORIG. The networks don't dial you up, so you must always be in the "Send" or "Originate" Mode. Be sure the DIR-ACP is set properly, according to the cable you're using. We'll assume you're using the direct connect modem cable in this chapter, so your switch should be set to DIR.

Using CompuServe

CompuServe Information Service (CIS) is one of the nation's largest and most well known information networks, and it is one service through which you gain access to many databases. You pay an hourly fee to CIS (\$5-\$20 per hour, depending on the level of service you order), and you can then choose from hundreds of different types of information banks. More and more companies are signing up their databases to be accessible through CIS. Occasionally, you may have to pay a small additional fee to get into a particular database, but you'll be told beforehand if you're to be charged. CompuServe can be reached in most cities by a local call. One of the sheets that come with the user's guide is a list of local numbers for major cities in the United States.

In addition to some of the data mentioned earlier, CompuServe also provides the following:

Electronic mail — send messages to any other CIS user nationwide.

Online encyclopedias — dial up information on any topic. This is an invaluable feature for students.

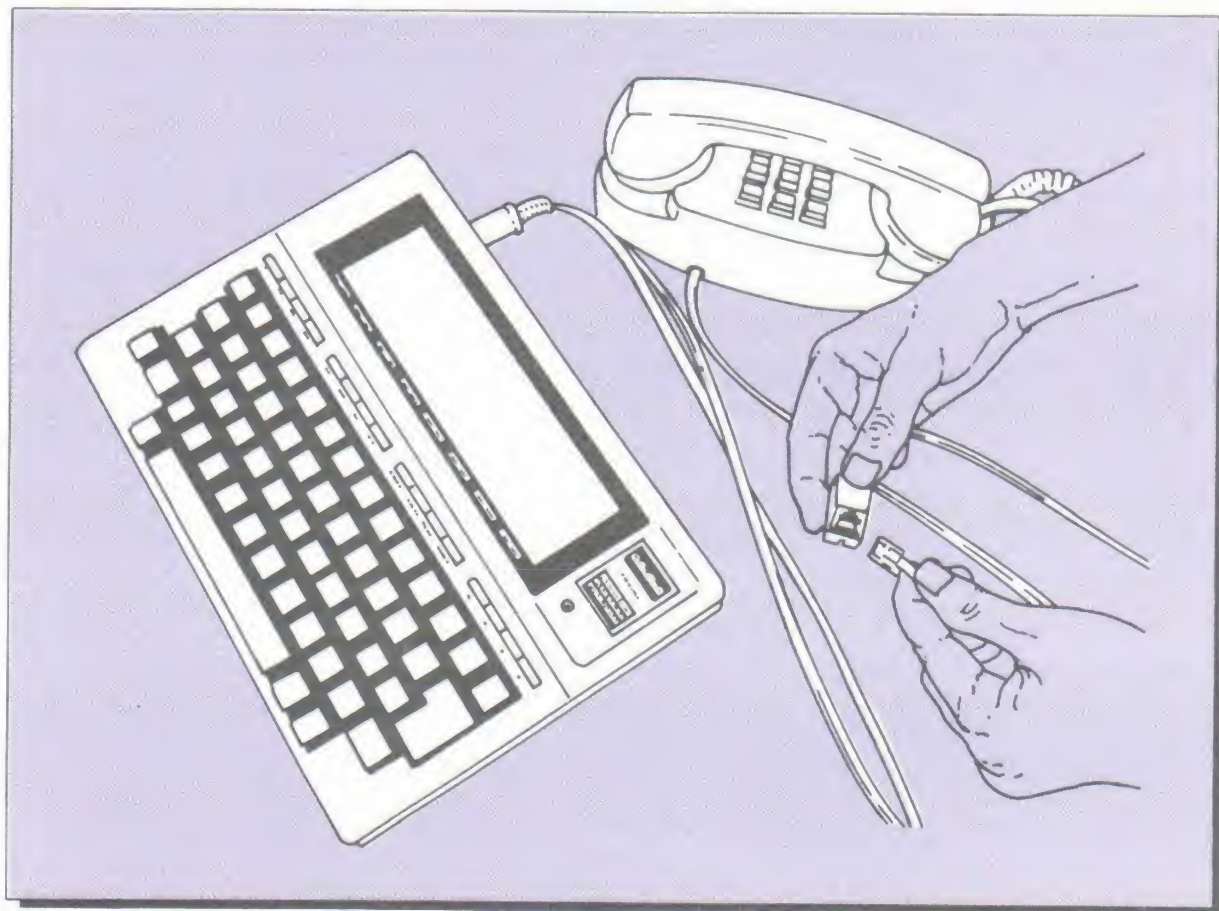


Figure 10-1. The Model 100, telephone, and modem cable

Home information — obtain electronic publications of information on a variety of topics, including gardening, nutrition, automobile maintenance, and many more.

Free software — download software from the Personal Computing Services.

Disk space — upload your Model 100 files to your private disk storage space. You can store as much as 128K.

Games — play your favorite game, from backgammon to *Adventure*.

In the following pages, we'll show you how to dial up CompuServe and how to use a few of its vast array of services.

Getting Started

Open your modem cable package and find the *CompuServe® Information Service User's Guide*. Included in this package is your *user identification number* (I.D.) and your *password*, both of which you need to *log-on* (connect) to CIS. You will have one hour of free connect time, during which you can subscribe to the service if you choose to do so. You don't have to stay on the entire hour at one session. CIS keeps a cumulative total of your time online. Also included in this packet is a list of CIS access numbers. Most users can dial a local telephone number to access CIS — no long-distance telephone charges.

Note that there are two basic types of services offered. *Regular* service is from 6 P.M. to 5 A.M. Eastern Time. *Prime-time* service is more expensive and is offered during business hours (from 5 A.M. to 6 P.M.). Your free hour is for regular service hours.

CompuServe offers a wide variety of databases to choose from, and, at first, it may seem like an impossibly complex system to deal with. It's not, really, and it's especially easy if you keep track of where you went in the databases and how you got there. Look over the *User's Guide* and plan what you want to do *before* you get online. Also, it's a good idea to write down your entries or print out a hard copy by using (F5), "Echo". You'll save a lot of time this way and get more out of your free hour — indeed, out of all the time you spend on CIS.

We'll show you a step-by-step example, just to get you started. Many people become confused when they first log-on to a system as large as CIS, so reading through this exercise and *then* trying it will make your initial time on CIS more enjoyable.



Hooking Up


For a detailed explanation of how to hook up your Model 100 to the phone lines, see Chapter 9. Here's a summary of the necessary steps:

Connecting the Model 100 to the Phone Line

1. Connect the cylindrical jack on the the modem cable to the Model 100 port labeled Phone.
2. Plug the silver wire into the back of a touch-tone telephone.
3. Plug the wire you took out of the back of the phone into the connector on the end of the beige cable wire.
4. Be sure your switches are set to DIR and ORIG.
5. Pick up the receiver and listen for a dial tone to confirm that you are properly connected.

Dialing Up

Once you've hooked up the the phone line, you are ready to use the TELCOM program. Use the  key to position the cursor over TELCOM and press . Be sure that the ANS-ORIG switch is set to ORIG and that the DIR-ACP switch is set to DIR (assuming you're using the direct connect cable).

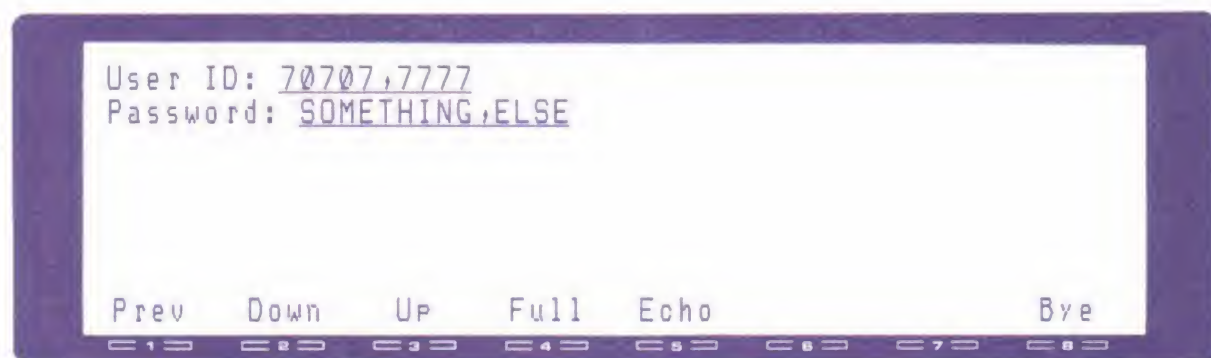
You are now in the *Entry Mode* of the TELCOM program. If you have entered your local CIS access number in your ADRS.DO file, you can use TELCOM's autodial feature, or you can enter the number from the keyboard. To enter the number from the keyboard, press  and enter the number:



Remember to pick up the receiver while TELCOM is dialing the number. As soon as dialing is complete and you hear that the line is open, enter the Terminal Mode by pressing (F4). You'll hear the Model 100 dialing, followed by a high pitched beep when the connection has been made, and the function key definitions at the bottom of the screen will change.

Logging On

Once the CompuServe computer has answered, the first thing you must enter is (CTRL) C (hold down the (CTRL) key and at the same time press C). Now you'll start getting prompts from CompuServe to enter your "User I.D.:" and your "Password:". Refer to the materials in your *User's Guide* to find these. Enter each as you are asked for them:



Your password will not show on the screen as you type it in. This is done for your protection. This way, if someone is watching you dial up CIS, he or she won't see your password on the screen.

The Main Menu

Now CompuServe is on the screen. The very first time you dial up, you will be given an opportunity to subscribe to CIS. The screen tells you exactly how to respond. The *User's Guide* also explains the questions you will be asked. Signing up is easy, and non-prime-time use costs about \$6 an hour.

After you go through the initial prompts about subscribing to CIS, you'll see the screen that will usually greet you. You will be shown a bulletin about

“What’s New” on the system. This is changed constantly, so what your screen actually says at this point will be different from what is shown here:

What’s New

New Service from News-A-Tron
New Access Numbers
Boston Access Number Changes
Aunt Nellie’s CB Conference

For details, See WHAT’S NEW
GO NEW at the ! prompt

Key <ENTER> for next page!

If you want a record of everything that’s passing before your eyes, you can make a hard copy by pressing function key **(F5)** to “Echo” the contents of the screen to the printer. You must be properly hooked up to the printer, of course. (See Chapter 7 for details of printer hookup.) Or, you can jot down the important details of what you’re doing as you go through the various databases.

After you read “What’s New”, you can get to the main menu by pressing **(ENTER)**. This and all other menus that appear on your screen may not be identical to the ones we’ve shown in this book; CIS is constantly updating their services, and, as they do, menu selections change, too.

CompuServe Page CIS-1

CompuServe Information Service

1 Home Services
2 Business & Financial
3 Personal Computing
4 Services for Professionals
5 User Information
6 Index

Enter your selection number,
or H for more information!

You select the section of CompuServe you want to enter by typing in its corresponding number on the menu and pressing **(ENTER)**.

But wait a minute! With the eight-line screen of the Model 100, these pages are scrolling off the screen. CompuServe provides a set of commands that allow you to stop this scrolling and to perform other functions on the screen.

CompuServe Commands

The first set of CompuServe commands we'll look at are called the *control character* commands, which you enter by simultaneously holding down the **CTRL** key and pressing a character key. You can stop scrolling, for example, by pressing **CTRL** S or **CTRL** A and resume by typing **CTRL** Q. Below is a summary of these commands:

CompuServe's Control Character Commands

- CTRL** C Interrupts the display (or program execution) completely so you can enter another menu selection or command.
- CTRL** U Deletes the line you are currently typing.
- CTRL** V Redisplays the partial line you are typing and allows you to continue typing the line.
- CTRL** H Backspaces, deleting the character that was on the screen. This character may not disappear from the screen, but it is no longer recognized by the computer.
- CTRL** A Temporarily suspends output at the end of the current line. (Enter **CTRL** Q to resume.)
- CTRL** S Temporarily suspends output immediately, even in the middle of a line. (Enter **CTRL** Q to resume.)
- CTRL** Q Resumes output after **CTRL** S or **CTRL** A.
- CTRL** O Interrupts output, which cannot then be resumed.

There is another set of commands you'll use on CompuServe — single-letter commands. You can view them by entering H **ENTER** at the main menu:

Enter your selection number,
or H for more information! H

T (TOP) returns you to the main menu
M (MENU) returns you to the previous menu
F (FORWARD) moves you forward one page
within the current series of pages
B (BACK) moves you back one page
within the current series of pages
H (HELP) displays information about the
area being used
R (RESEND) resends the current page
S (SCROLL) displays pages continuously
without stopping for the prompt at the
end of each page
N (NEXT) selects the next item from
the most recently used menu without
redisplaying the menu

The control character commands can be typed while CompuServe is in the process of transmitting to you, but these single-letter commands can be used only when CompuServe is waiting for you to give it instructions. You'll know that this is the case when you see the CompuServe prompt on the screen. This prompt is an exclamation point (!).

Now you're ready to have some fun. With these basic commands, you have CompuServe at your fingertips.

After the Main Menu

As you probably realize, there are a great many things you could do now. As an example, we'll take you through a brief, step-by-step explanation of how to read reviews of some current movies.

Notice that the upper right corner of every CompuServe screen (except the "What's New" screen) has a page number. The main menu is page CIS-1. As mentioned earlier, you can get to a new menu or new area within CompuServe by entering the menu-selection number. However, once you know the page number of the topic you want, you can simply type GO, followed by the page number. You could, for example, type GO HOM-1 **ENTER** from the main menu to get to the first page of "Home Services".

For this example, we'll choose the option for "Home Services". Press the number 1 and then **(ENTER)**, and you'll see the first "Home Services" menu:

```
Enter your selection number,  
or H for more information! 1
```

```
CompuServe          Page HOM-1
```

HOME SERVICES

```
1 News/Weather/Sports  
2 Reference Library  
3 Communications  
4 Home Shopping/Banking  
5 Discussion Forums  
6 Games  
7 Education  
8 Home Management  
9 Travel  
10 Entertainment  
Last menu page, Key digit  
or M for previous menu,!
```

The list shown is longer, of course, than that on the Model 100 screen, so it partly scrolls off the top. Use **(CTRL)** A to stop the scrolling. It might be a good idea to print out these lists, at least in the beginning, so you can see the complete menu selections all at once.

You will find movie reviews under the "Entertainment" section, so choose selection 10. (It may be a different selection number on your CompuServe screen.)

```
Last menu page, Key digit  
or M for previous menu! 10
```

```
CompuServe          Page HOM-31
```

ENTERTAINMENT

```
$ 1 Hollywood Hotline  
2 Rick Brough's Movie Reviews  
3 Video Information  
4 SHOWTIME Movie Catalog  
5 Prime-time Radio Classics
```

\$ Indicates surcharged service

```
Last menu page, Key digit  
or M for previous menu,!
```


You can see that there is a wide variety of entertainment news here. Selection number 1, "Hollywood Hotline", will charge an extra fee (in addition to the hourly CompuServe charge) if you choose that option. You cannot enter any of the "pay" areas of CompuServe during your first, free hour. Once you've signed up, however, you can go almost anywhere in the various database areas.

Let's choose the option for "Movie Reviews", selection number 2:

```
Last menu page, Key digit  
or M for previous menu! 2
```

```
CompuServe          Page NMM-1
```

```
* RICK BROUGH'S MOVIE REVIEWS *  
  are published weekly in  
"The Park Record," Park City, UT
```

```
1 MOVIE RATING GUIDE  
2 Ghostbusters  
3 Temple of Doom  
4 El Norte  
5 Star Trek IV  
6 Rope  
Input a number or key  
<ENTER> for more choicesY,=!
```

Now you are given the option of selecting from different movies that were recently released. Choose the movie you'd like to read about. When you are done, you can return to the CIS main menu by typing in T ENTER.

Saying Goodbye

When you are finished with CIS, you can hang up by typing BYE at CIS's prompt (!). Then press the function key labeled "bye", F8. You'll then be asked if you really want to hang up with the prompt "Disconnect?". Say "Yes" by pressing Y, followed by ENTER. If you disconnect without typing in BYE, CIS may not know you've hung up and may continue charging you.

```
6 Rope  
Input a number or key  
<ENTER> for more choicesY,=!
```

```
Disconnect?Y  
Prev Down  Up  Full Echo      Bye
```

You are now disconnected from CIS. If you decide to join, you can order reference manuals for the service by choosing "User Information" from the main menu (page CIS-1). Then find the option for "Feedback, Manuals, and Products", and you're on your way!

Other CompuServe Services

CompuServe is growing larger and larger every day. It would be impossible to give an exhaustive account of every area of the service, but two of the most popular are "Electronic Mail" and the "Special Interest Groups" (SIGS).

Electronic Mail

"Electronic Mail" (EMAIL) is a service that allows you to send a message to another CompuServe user *anywhere in the country*. You must know the User I.D. of the person you want to send a message to. There is an index of CompuServe User I.D. numbers under the "Index" section on the main menu (CIS-1).

You can compose a message on your Model 100 in the TEXT program and send it via EMAIL. You must sign up as a permanent CompuServe user to access EMAIL. You'll find detailed instructions in the *User's Guide* that you order through the "Manuals" section under the selection for "User Information" on the main menu.

Special Interest Groups (SIGS)

A "Special Interest Group" (SIG) is a unique area of CompuServe that allows users with similar interests to form an "electronic club". The purposes of SIGS range from specific computer hardware to cooking. The Model 100 SIG is one of the most active on CompuServe.

Every SIG has a system operator (SYSOP) who maintains all the messages and supervises the general flow of information within the SIG. The SYSOP may charge dues for membership, but most SIGS are free (except, of course, for the hourly CompuServe fee). Each SIG consists of "Electronic Mail", "Conferencing", and "Special Databases", from which you can download programs. (See the instructions for downloading in Chapter 9.)

You can get a list of the computer SIGS by choosing the selection for "Personal Computing" from the main menu. If you want to access the Model 100 user's group, choose the selection for "Groups and Clubs", then choose the option for the Model 100 SIG.

Using Dow Jones

Dow Jones and Company has been bringing financial news to folks for a long time. The company was formed in 1882, and today it includes a multitude of services, including the largest selling daily, *The Wall Street Journal*, and one of the best financial information services in the country, the *Dow Jones News/Retrieval* (DJNS) service.

With the Model 100, the Dow Jones News/Retrieval, like CompuServe, is right at your fingertips. A user's guide and one hour of free connect time is included for DJNS in your Model 100 modem cable package.

The *Dow Jones Information Service User's Guide* that comes with your modem cable will help you log-on. We won't go into detail about that here because it is in many ways similar to the procedure for CompuServe. You'll find a complete picture of the Dow Jones service in *Practical Finance on the TRS-80® Model 100*, by S. Venit and Diane Burns (New York: Plume/Waite, New American Library, 1984).

One thing that's important to know, and the manual doesn't tell you this, is that *you must change the STATUS of your TELCOM program to dial up DJNS*. The proper setting is: M7I1D. This changes the line status from Enable to Disenable.

Refer to this chapter and the previous one on changing the status and on hooking up and dialing DJNS. Then use the User's Guide to log-on and explore the many available databases.

Coding an Auto-Log-On Sequence

One of the nicest features of the Model 100's TELCOM program is that you can log-on to an information service automatically. By coding in your User I.D. and password information just once in a special format, you no longer have to go through the relatively tedious motions of logging on each time you dial up. After all, computers are terrific at handling these tedious, repetitive tasks — so why not let them?

We'll show you an auto-log-on sequence for the CompuServe example described above. By following the guidelines given here, you can automatically log-on to any other computer. You can even write the sequence to get you to a specific location within the information services, a great help if you are repeatedly going to use the same databases.

The auto-log-on sequence must always be entered in ADRS.DO, after the name and number. You *must* have a direct connection modem cable to execute an auto-log-on sequence.

Below are some codes you'll need to know to use the auto-log-on feature.

Coding an Auto Log-On Procedure

Command	Function
---------	----------

<>	Entire sequence must be inside these brackets
=	"Pause for two seconds"
?	"Wait for a specified character"
^	"The following character is a control character"
!	"Send a specified character"

Our auto-log-on sequence for CIS, entered in ADRS.DO, looks like this:

```
CIS :9561000 <=^C?U70707,7777^M?PSOMETHING,ELSE^M>
```

Don't be discouraged if this seems like so much Greek. Each letter in the sequence has its own simple meaning. Let's take them one at a time:

CIS :9561000 Name and number you're calling (see the chapter on ADRS.DO for auto dial procedures).

< Begin auto log-on sequence.

= Pause for two seconds, giving the host computer time to establish a good connection.

^C Sends a control C to CIS (get the "^" character by typing SHIFT 6).

?U The question mark means wait for the next character, in this case, the "U" from CIS's prompt "User ID:".

70707,7777 Since you were prompted for your User ID in the previous step, the Model 100 sends it over the line to CIS, just as it appears here.

^ M This part of the sequence sends the equivalent of pressing the **ENTER** key, necessary after typing in your ID.

?P The question mark means wait for the next character, in this case, the "P" from CIS's prompt "Password:".

> End auto log-on sequence.

Type this sequence in your ADRS.DO file. When you're ready to log-on, follow the usual beginning procedures:

1. Connect your Model 100 to the telephone lines.
2. Turn on the Model 100 and enter TELCOM.
3. Press **F1**, "Find", and enter "CIS".
4. Press **F2**, "Call"; you may need to pick up the receiver during dialing.
5. When dialing is complete, press **F4** to get into Terminal Mode.

That's all there is to it! You can just sit back and relax as the Model 100 does the work for you.

On Your Own

Explore the "Home Services" section of CompuServe (from the main menu). Try getting a weather report or go to the "Games" section and play *Hangman*.



Hookup to Other Computers: The RS-232 Port

Concepts

- What you can do
- What you need
- Saving and loading text files
- Saving and loading BASIC files

As we've seen in the previous chapters, there are many reasons why the Model 100 has enjoyed such tremendous success. It is a truly portable machine, and it can be used to "talk" to computers and information services all over the world. But the Model 100 has another important capability: it can operate as an extension of your full-size microcomputer.

Through the RS-232 port you can hook up the Model 100 to any computer in your home or office. You can write a memo on the Model 100 and then transfer it to another microcomputer, where you can store it on a disk, use the word-processing package on your micro to complete the final formatting or stylistic touches, or print out the document. Or, you can write a BASIC program on your full-size computer, making use of its full-screen display and sophisticated debugging aids, and then transfer the program to your Model 100 for use in the field.

All this and more is possible, thanks to the RS-232 port on the back of the Model 100. Don't feel that this mysterious-sounding "RS-232" is over your head; it's really not difficult to understand. The RS-232 port is simply a channel through which two computers communicate with one another; it

is also known as a “serial” port, because it transfers data serially, or one bit at a time. The RS-232 serial port is accessed through the TELCOM program. How to do this is the subject of this chapter.

What You Need

To perform a file transfer, your microcomputer, like the Model 100, must have a serial port. Some microcomputers come with one; others do not. It's also a good idea to have a telecommunication software package for your microcomputer, one that will perform similarly to TELCOM.

In addition, you will need two other items (see Figure 11-1). One is an RS-232 cable, also known as a DB-25 connector. The 25-pin configuration of the RS-232 port is a standard in the electronics industry, and many electronics stores sell these cables. Cables are available in varying lengths; the important thing is that you get a cable with male-to-female connectors (like RS #26-4403). The second item you will need is the *null modem adaptor* (RS #26-1496). This adaptor is necessary because both your microcompu-

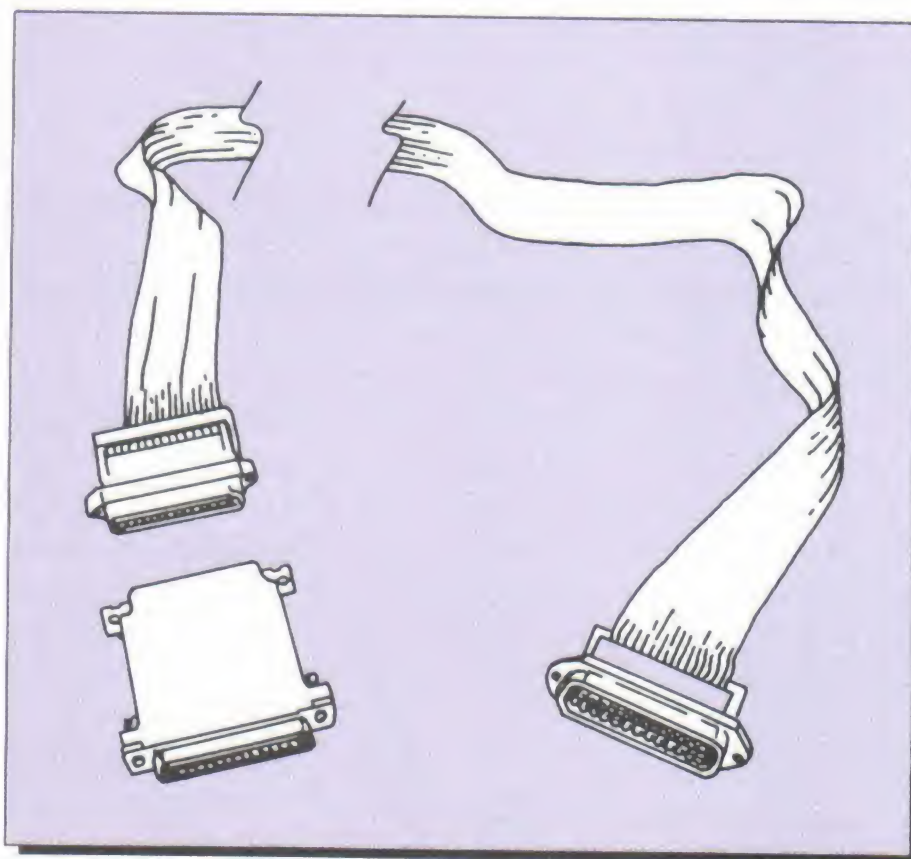


Figure 11-1. The RS-232 cable and null modem adaptor

ter and the Model 100 send and receive signals through the same pins in the RS-232 serial port; that is, they share a single line in the cable. To avoid a crossed connection, the null modem adaptor switches the sending and receiving pins, so the Model 100 receives through the channel the microcomputer is sending through, and vice versa.

Hookup

First, connect the null modem adaptor to the RS-232 serial port on your microcomputer. The null modem adaptor will not fit directly on the Model 100. Notice that the connector can fit in only one direction. Next, take the female end of the RS-232 cable and connect it to the null modem adaptor. Then take the male end and plug it into the port on the back of the Model 100 labeled RS-232C (see Figure 11-2). That's all there is to the hardware part of the RS-232 connection.

Saving and Loading Text Files

After you have properly connected the Model 100 to your microcomputer, you must set up both computers so that their software speaks the same language. Just as in the previous chapters, where we described communication over the phone lines, the communication parameters in the Model 100 TELCOM program must match those in your microcomputer's telecommunication software. You can choose the settings you like, but they must be the same on both computers.

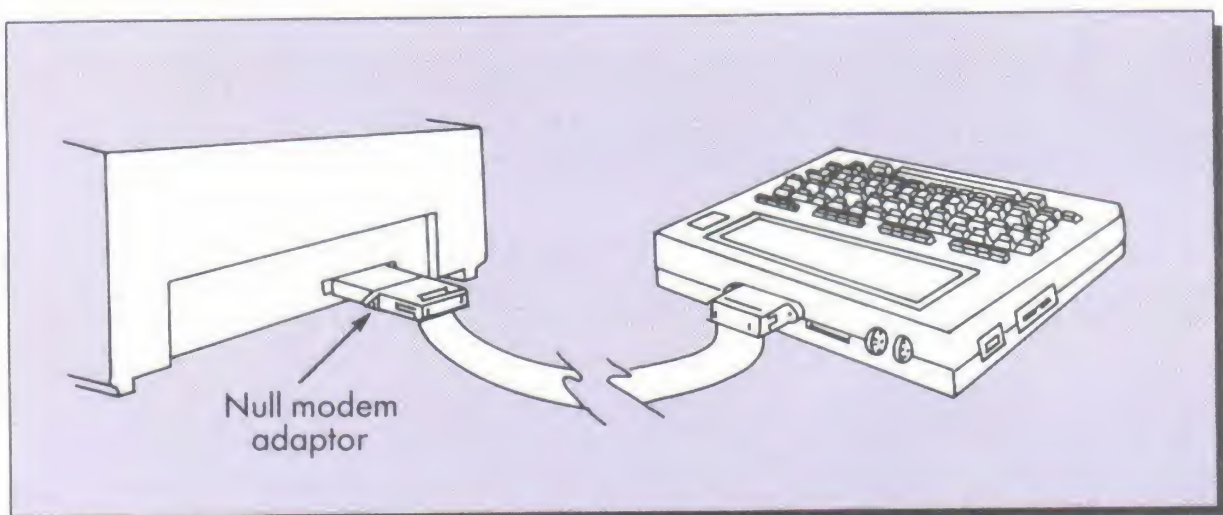
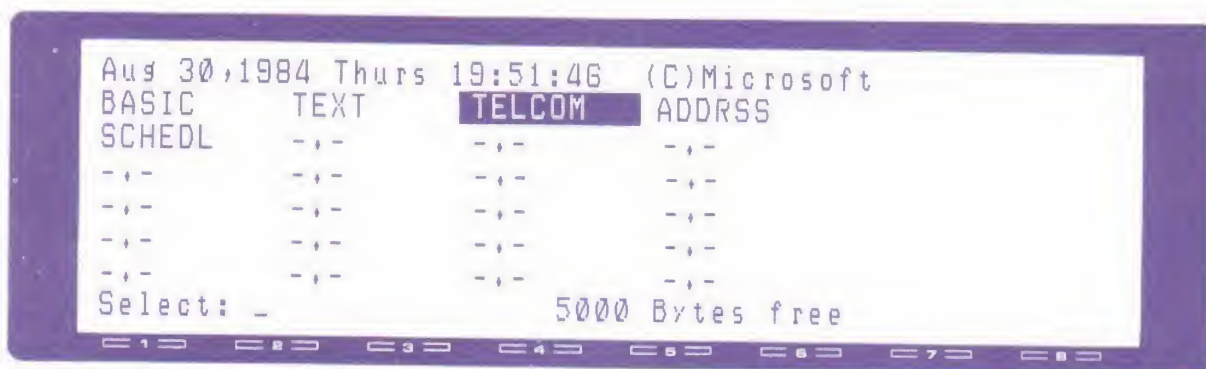


Figure 11-2. The Model 100 — microcomputer connection

The examples used in the following sections were generated on an IBM PC, using the telecommunication package CROSSTALK XIV.

In order to set the parameters on the Model 100, position the cursor over TELCOM and press **(ENTER)**.



You are now in the Entry Mode of the TELCOM program. Remember, unlike the other programs in the Model 100, TELCOM actually has two different modes of operation — the *Entry* Mode and the *Terminal* Mode — and the function keys have different definitions in each. We'll use the Entry Mode to set our communication parameters and the Terminal Mode to do the actual "talking" to the other computer.

You'll recall from Chapter 9 that the communication parameters are changed by using the STAT function, so press **(F3)**. The parameters are currently set for use with the modem, so we'll have to change them. In our particular case, the new settings are 57N1D. (You can read more about these settings in Chapter 9.)

Here is a summary of what each number and letter stands for:

- 5 "The speed of the transmission will be 1200 bits per second (1200 baud)"
- 7 "Each character has seven bits of basic data (data word)"
- N "I am not using any parity for error checking"
- 1 "Each character will have one bit to signal the end of that character"
- D "I'm not going to send in XON/XOFF protocol"

After you enter the new parameters, press **(ENTER)**.

To view the settings to see if they're correct, press **F3** **ENTER**. Your screen should look like this:



Load your telecommunication software disk in your microcomputer and set its parameters to match those of the Model 100. In this example, we used the following settings in CROSSTALK on our IBM PC:

SPeed	1200 baud
DAta bits	7
PArity	None
STop bits	1

You enter these commands in CROSSTALK by pressing **ESC**. The **ESC** key puts CROSSTALK in Command Mode. (Of course, you should consult the manual for the equivalent commands in your particular telecommunication software.) Next, you enter the command, using the first two letters of the command. Then the proper setting is entered.

We also used CROSSTALK to put our microcomputer in Answer Mode and set it up so that it will write all incoming data to disk:

MOde	Answer
CApture	To the file on disk B:STORY.100

Saving Text Files to Disk

Now you're ready to transfer files between the Model 100 and your microcomputer. Let's say that you've written the great American short story on the Model 100. Now you want to transfer the file to your microcomputer so that you can improve the format before sending it off to a publisher.

Remember, TELCOM has two modes. You were in the Entry Mode to change your communication parameters; now you must get into Terminal Mode in order to send your short story to the microcomputer. Change into the Terminal Mode by pressing **F4**. As soon as you enter the Terminal

Mode, you will probably get a message from the telecommunication software on your microcomputer, letting you know that it's ready to receive data.

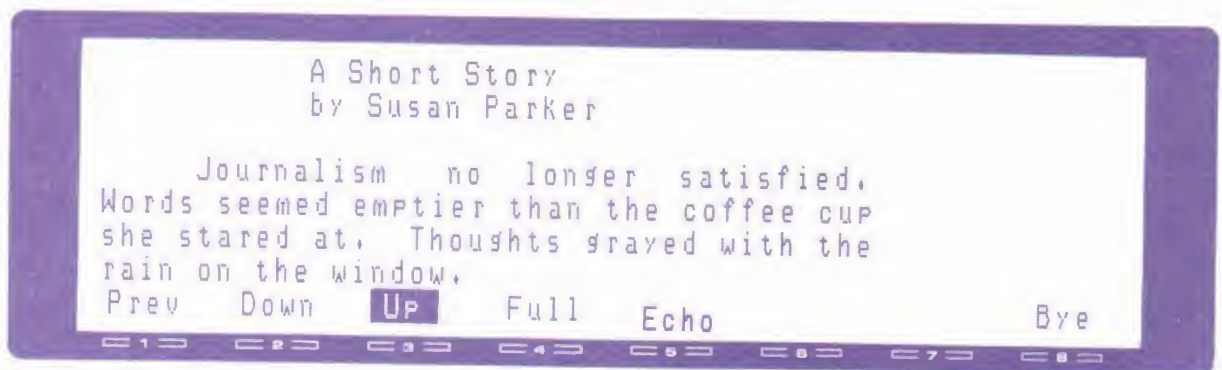
Now you're ready to send the file containing your short story. In order to send a file, you must *upload* it to your microcomputer. Press (F3), and the screen prompts "File to upload?". Simply enter the filename:



Now TELCOM asks for the width. You may enter any number between 10 and 132, indicating how many characters the Model 100 will send before it sends a carriage return. The word-processing package for the IBM PC uses a width of 65.



Two things happen at this moment: the word *Up* at the bottom of the screen appears in *reverse video*, and the file begins scrolling across the screen.



The entire file will scroll across the screens of both your Model 100 and your microcomputer, and you should be able to see the microcomputer's disk drive light up every so often as the file is being written to the disk.

You'll notice that all this happens quickly. Just how quickly depends on the baud rate you set using STAT. You can set the Model 100 to transmit to speeds as high as 19,200 baud (19,200 characters per second). Of course, the telecommunication software in your desktop computer must also be able to send and receive data at that speed.

When the file has been completely transmitted to the microcomputer, the scrolling will stop and the Up label will no longer be in reverse video.

At this point, you could send another file to the Model 100. If you decide to do that, you must change the file your microcomputer is saving the data in, unless you don't care if two files get saved under one name in the microcomputer.

Once you've saved a file to disk, you may delete it from your Model 100. Return to the Entry Mode of the TELCOM program by pressing (F8) and tell the Model 100 you want to disconnect by pressing Y (ENTER). Now return to the main menu by again typing (F8).

To disconnect your microcomputer through CROSSTALK, press (ESC) to get into the Command Mode and type QUIT (ENTER). You can now delete the file on your Model 100 by typing

```
KILL "SHORT.DD"
```

Now the file is stored on your microcomputer's disk, ready for you to forge ahead in your writing career.

Loading a Text File from Disk

If, however, you've been working on a history text and want to spend a day at Gettysburg to gather firsthand impressions to add to your chapter on the Civil War, you can transfer a file from your microcomputer's disk into the memory of your Model 100.

The steps are similar to those described in the previous section. Be sure you've used the RS-232 cable and null modem adaptor to properly connect the Model 100 to your microcomputer's serial port. Also, be sure there's enough memory in your Model 100 to hold the file you're about to download.

Use the arrow keys to place the cursor over TELCOM and press (ENTER). Remember, you must set the communication parameters on your Model 100 to be the same as those on your microcomputer, as described in the previous section. Now, when both machines are set, change to the Terminal Mode by pressing (F4) and begin downloading by pressing (F2). The prompt "File

to Download:" will appear on the screen. Enter the filename you want the text to have on the Model 100 — not necessarily the filename you've used for it on disk.



Suddenly the word *Down* is in reverse video:



Now you must enter a command (often *SEND* or something similar) to your microcomputer to send the file from the disk. As soon as the command has been issued, you will see the file going across the screen on your Model 100. Each line will overwrite the previous one on your Model 100 display, because the Model 100 inserts its own line feed after the information is shown on the screen. The file is being received successfully, however, and will look fine once it's in your Model 100. The file will also be scrolling across the screen of your microcomputer, where it will look normal.

When the transfer is complete, the scrolling will stop. The download function will not turn off on its own, however; you must end it manually by typing in (F2).

Exit from the Terminal Mode of TELCOM by pressing (F8). When you see the prompt "Disconnect?", type Y (ENTER); then return to the main menu by pressing (F8).

Look at your main menu. There, among the other files, you'll find "HISTORY.DO". Place the cursor over the filename and press **ENTER** to view the file, using TEXT. Now you can edit your Civil War chapter, filling in your firsthand description of the rolling Pennsylvania countryside.

Saving and Loading BASIC Files

In addition to text files, you can also transfer BASIC programs between the Model 100 and your desktop computer. Be aware, though, that the BASIC programs you write interact directly with the particular machine they're running on. The program that runs on the Model 100 is not necessarily right for your IBM PC, or even for your desktop TRS-80 microcomputer. Almost all the BASIC programs you transfer from or to another computer will require some "tinkering" before they will run.

There are, however, many good reasons for transferring BASIC programs between computers. The most obvious is that you've written a program on one machine that you'd like to run on another one. If it's a long program, you can save a lot of time by transferring it via an RS-232 cable instead of typing it in again. Or, as we mentioned earlier, you may want to take advantage of the superior editing capabilities of a larger computer, even though you're writing a program to run on the Model 100.

Connecting Up

Connect your Model 100 and your microcomputer by following the steps outlined in the section on hookup. Then set the communication parameters on both computers to match each other. Don't forget to set your desktop microcomputer to the Answer Mode, ready to save incoming data to disk. On the Model 100, use the STAT function in the TELCOM program. All the instructions up to this point are the same as in the sections on text files.

Sending BASIC Files from the Model 100 to Another Computer

Now, instead of going into the Terminal Mode in TELCOM, you need to be in BASIC. If you are still in TELCOM, make a note of the STAT setting you have entered (for example, 57N1D). Then exit TELCOM by entering **F8**.

From the main menu, enter the BASIC program.



Load the file you want to transfer into the BASIC work area by pressing (F2). When the prompt "Load:" appears, enter the filename, PROG, for example.

To send the file through the RS-232 port, press (F3), and, when the prompt "Save:" appears, enter "COM:", followed by the appropriate Stat settings:



The parameters you enter in this command must be the same as those to which your TELCOM program is currently set.

You will not see the file scrolling across your Model 100 screen, but the file will appear on the screen of your desktop microcomputer. When the transfer is complete, the file will stop scrolling. You can then return to the main menu by pressing (F8).

Loading BASIC Files from Another Computer

As you may have guessed, loading a BASIC file from your microcomputer to the Model 100 is going to be as easy as pie.

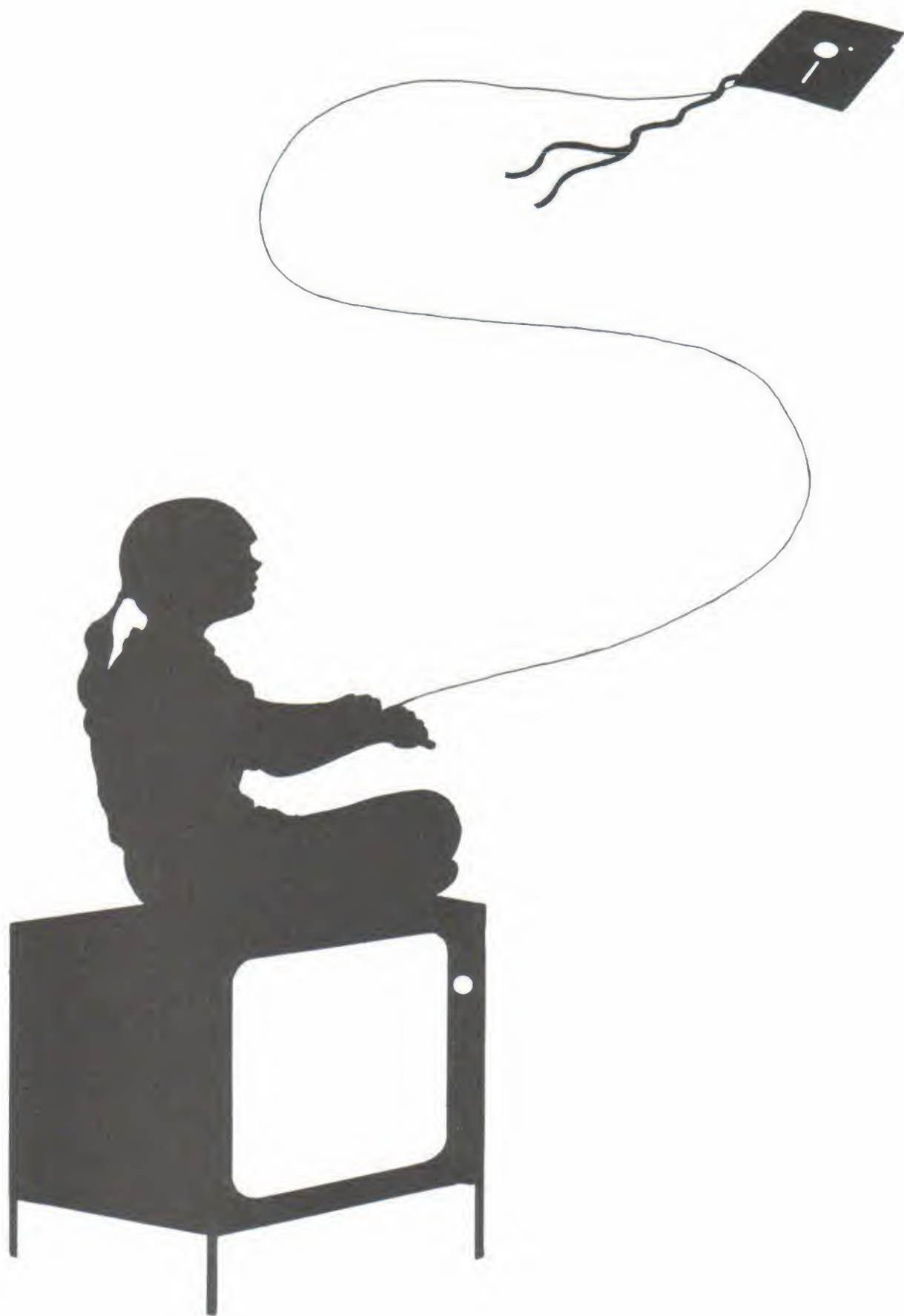
With the communication parameters properly set on both machines and your microcomputer set up in Answer Mode, enter BASIC on your Model

100. Press **(F2)**, and, when the screen asks what file you want to load, enter the name you want the file to have on the Model 100: **PROG**. (Remember, there is a six-letter limit on program names.)

Now the Model 100 is set to receive the file from your microcomputer. You need only enter the command to your microcomputer that will cause it to **SEND** a file. The file will scroll across the screen of your microcomputer but not your Model 100. When loading is complete, the file will stop scrolling. You may run or list the program, but be sure to save it before you leave **BASIC**.

Summary

We've shown you how to transfer files between your Model 100 and another computer, using an RS-232 cable. Although our examples made use of an IBM PC, you should have little difficulty adapting the procedures to other computers, using different telecommunication software.



12

Using the Optional Disk/Video Interface

Concepts

- Connecting the disk/video interface to the Model 100
- Starting up the disk/video interface
- Backing up the system diskette
- Saving and loading TEXT files to disk
- Saving and loading BASIC programs to disk

If you spend much time using the cassette recorder to save programs or data files, you are already aware of its limitations. It gets the job done, but there's an alternative which is faster, more convenient, and more reliable. For those of you who have struggled with the cassette recorder and wished for something better, Radio Shack has the answer: an option they call the *disk/video interface*. (We'll sometimes refer to it simply as the *interface*.) In this chapter, we'll describe the interface, and show you how to set it up and get it going, as well as how to use it to store and retrieve BASIC programs and files that you generate with TEXT.

What Is the Disk/Video Interface?

Disk drives are a much faster, more convenient, and more reliable way to store data than are cassette tapes. A single floppy disk can hold about 160,000 bytes — enough for dozens of programs or data files. You can access any of these programs or files in just seconds, with high reliability and with no more trouble than slipping a floppy disk into the slot in the disk/video interface.

In addition to disk storage, the Radio Shack people added a video interface, which means that you can use the screen of any TV set or video monitor using the Model 100 screen. Such screens are not only brighter and easier to read than the LCD (liquid-crystal display) screen on the Model 100, they also give you a larger viewing area.

The disk/video interface itself consists of a metal box about a foot square and five inches high. It has a built-in disk drive and space for an additional drive. There is a ribbon-cable connector for connecting the interface to your Model 100, and separate plugs for TV and video monitors. The interface runs on regular household current; the power switch is located, inconveniently, in the back of the unit (see Figure 12-1).

Though it's not large, the interface box is fairly heavy, much heavier than the Model 100 itself. You will probably want to set up the disk/video interface and a TV set, more or less permanently, on your desk. Then, when you want the advantages of the disk drive, you'll connect up your Model 100 to the interface, and, when you want the advantages of portability, you'll disconnect it and be free to roam the world with your Model 100.

The disk/video interface in effect gives you two computers in one: a true portable and a larger, more sophisticated, desktop model. In some ways, the arrangement is similar to that described in Chapter 11, where we showed you how to hook up the Model 100 to other computers. However, because the disk/video interface is built specifically to interface with the Model 100, it provides in many ways a more convenient setup, as you'll see.

Are there any disadvantages to the disk/video interface? Besides the fact that it is a fairly expensive unit, there are some things to look out for when connecting it up and using it, as you'll see in the following sections.

Connecting the Disk/Video Interface to the Model 100

In this section, we'll show you how to connect the disk/video interface to your Model 100.

First, remove the plastic cover from the *ROM module expansion compartment* on the bottom of the Model 100. It's located near the front, just under the keyboard. You can pry it up with a dime or a Swiss army knife. Underneath you'll find two different sockets. The one located closer to the front of the computer has 28 sliding connectors — it's *not* the one you want. The other connector has 40 little holes arranged in two rows of 20. It's lower and wider than the first connector and is located more toward the center of the computer (see Figure 12-2).

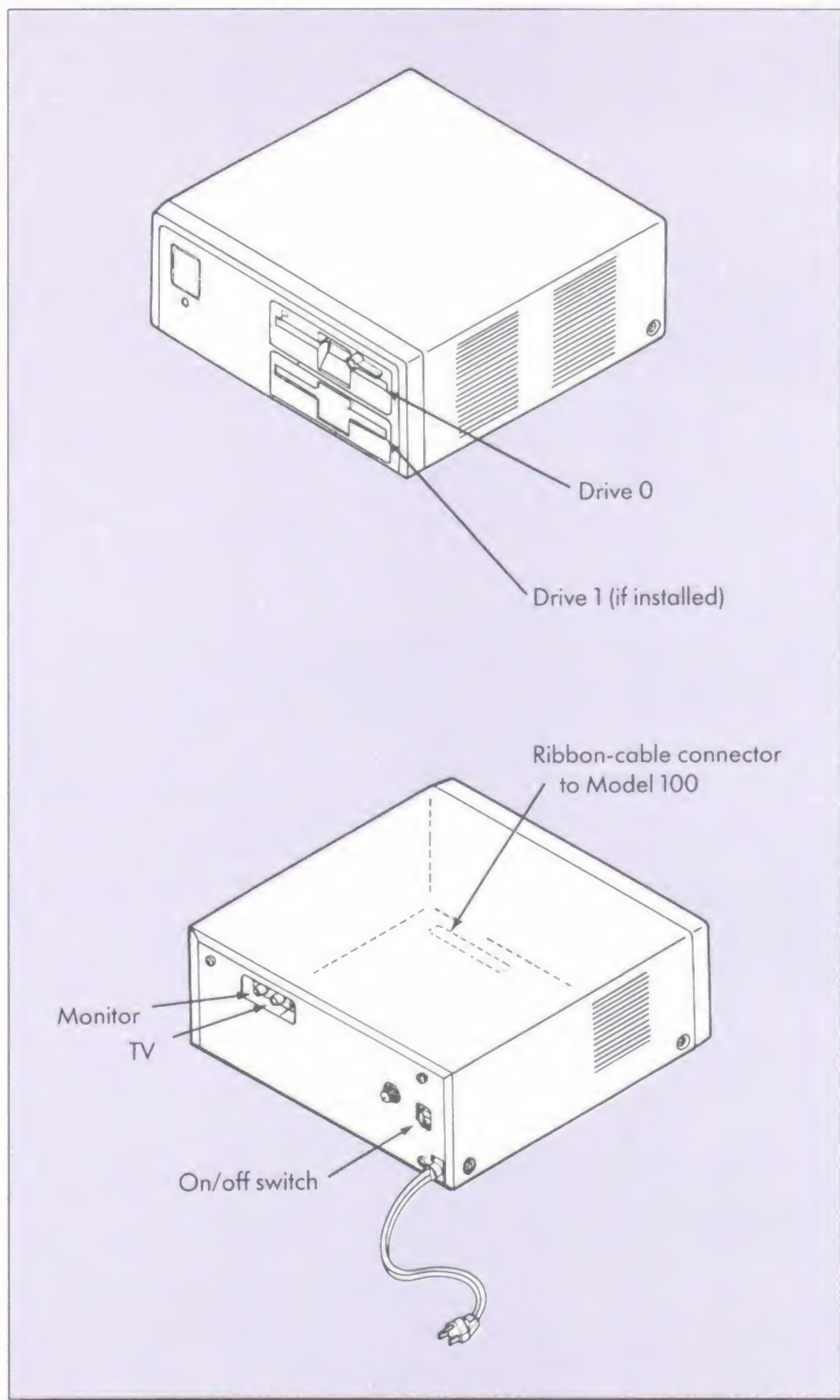


Figure 12-1. The disk/video interface

Now that you've found the right socket, it's time to plug in the cable to connect the Model 100 to the disk/video interface. The flat ribbon-type cable comes with the interface. On one end is the connector for the 40-pin socket on the Model 100; on the other is a different type of connector, which plugs into the interface. The end that goes into the Model 100 has 40 little wire pins sticking out. (The interface connector has 40 sockets arranged in a grid-like pattern; we'll ignore this end of the cable for the moment.)

If the disk/video interface has an Achilles heel, it is this 40-pin connector on the end of the ribbon cable. *The pins are very fragile*; it is dismayingly easy to bend them over when you try to plug the cable into the socket in the Model 100, or even when you are handling the cable. Even people experienced with electronic equipment have been known to bend the pins. Once the pins are bent, attempting to straighten them can result in breaking them off — bad news, because you will then need a new cable. We suggest that you have the people at Radio Shack show you how to make this connection. This may give you the expertise you'll need to attach the cable on your own. In any case, *handle the cable with care*.

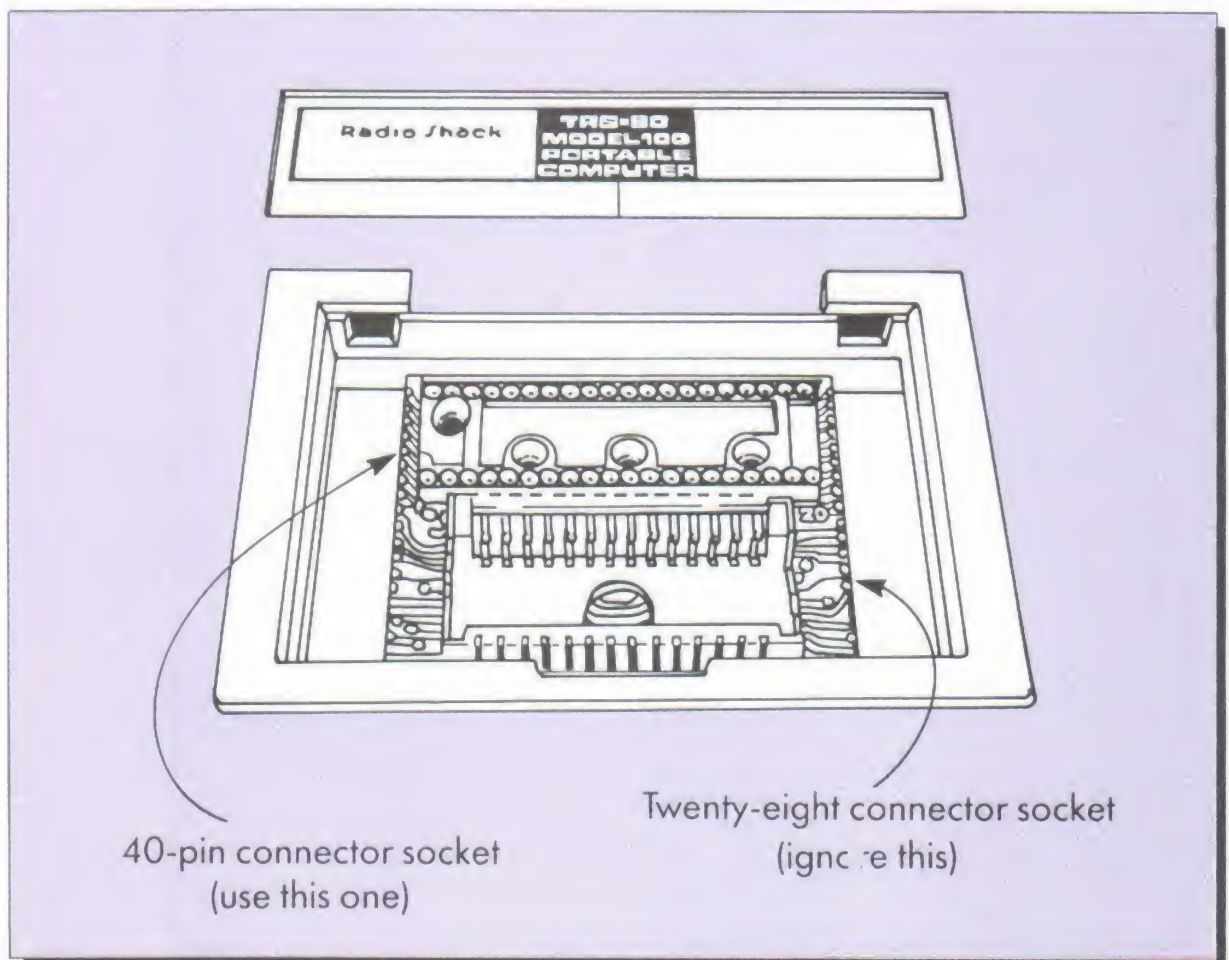


Figure 12-2. The ROM module expansion compartment

Plugging the Adaptor into the ROM Socket.

There are really two steps to attaching the ribbon cable to the Model 100. First, you plug a special adaptor into the ROM socket on the Model 100, and then you plug the cable into the adaptor. The adaptor serves two purposes: it makes it easier not to bend the pins on the cable connector, and it gives the socket a little extra height so that the cable can follow a more direct path out of the bottom of the Model 100.

The adaptor is a small plastic device with 40 pins on the bottom and 40 sockets on top (see Figure 12-3.) On the side is a little lever. When you raise the lever, the holes in the top of the adaptor become larger, so you can plug in the connector on the end of the cable. When you push the lever down alongside the adaptor, the holes close up, locking the pins of the cable connector in place.

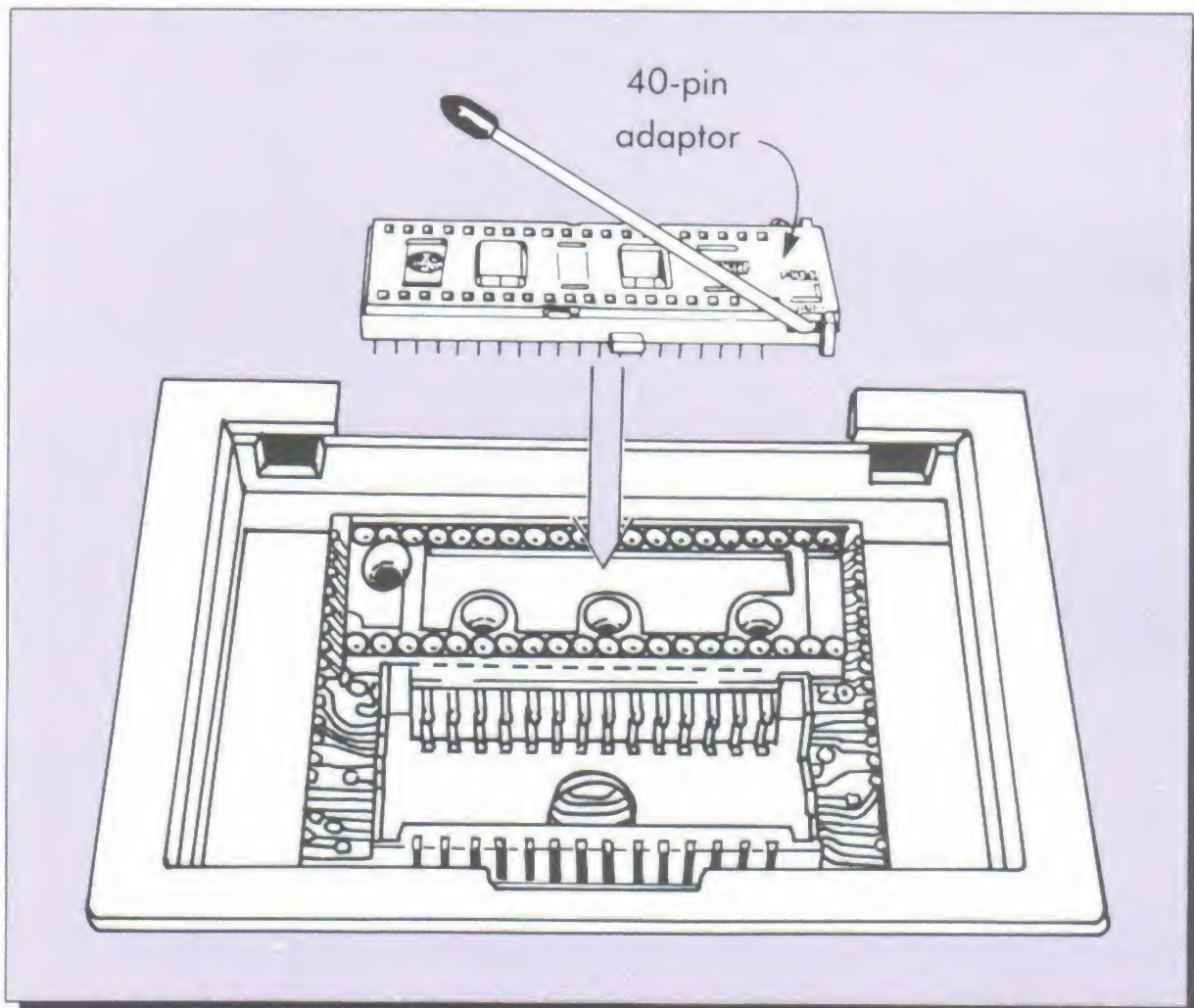


Figure 12-3. The 40-pin adaptor

The first step is to plug the adaptor into the ROM socket. If the adaptor has a piece of foam plastic protecting the pins, remove it. Check that all the pins are straight — if not, bend them very gently until they are straight and neatly aligned. Now, very gently and carefully, plug the adaptor into the 40-pin socket. The little lever on the adaptor should face toward the front of the Model 100, that is, toward the other socket. Press the adaptor into the socket slowly and evenly, so it sinks in on an even keel. If it's tilted when it goes in, the pins may be bent.

Plugging the Cable into the Adaptor

The cable is oriented so that it leads from the ROM module expansion compartment toward the back of the Model 100. This is because the disk/video interface sits *behind* the Model 100 on your desk.

Now comes the hard part. The pins on the cable connector are even more fragile than those on the adaptor. Raise the lever on the adaptor so that the holes open up. Now, very gently, evenly, and carefully, ease the pins on the cable connector into the holes in the adaptor. They should go in easily. Now, lower the lever so that it lies alongside the adaptor.

To finish off this part of the job, take the new plastic cover supplied with the interface, and snap it into place over the top of the ribbon-cable connector. It holds the connector in place and keeps dust out of the compartment.

Plugging the Cable into the Disk/Video Interface

The next step is easier, because the socket that connects the ribbon cable to the interface is more sophisticated than the socket on the Model 100.

Place the interface unit upside down on your desk, behind the Model 100. The Model 100 should also be upside down, with the cable leading from the Model 100 toward the interface. Now, plug the free end of the cable into the socket on the bottom of the interface box (there's only one socket here, so that's easy). As you push the connector into place, two little levers on the sides of the connector will move toward it, locking it in place. If you should ever want to remove the cable, simply push these levers away from the connector, and the connector will pop out of the socket.

Now that everything is plugged together, you can turn the interface and then the Model 100 rightside up.

Connecting the TV Set to the Disk/Video Interface

If you're going to use the disk/video interface, you'll need a TV set or video monitor. A *monitor* is a video display unit that provides a higher-quality picture than a TV set. It also costs more money. If you don't own

one already, you'll find that any color or black-and-white TV set will work fine. From now on we'll refer to a TV set, but a monitor will function in the same way.

You need a TV set because many of the messages that tell you what to do with the disk system appear *only* on the TV screen — not on the screen of the Model 100. Thus you won't know what the disk drive is doing if you don't have a TV set hooked up to the interface. TV sets are ubiquitous and inexpensive, so this shouldn't be a problem.

If you're using a monitor, simply plug its cable into the jack labeled Monitor on the back of the interface. If you're using a TV set, you'll need something called an RF adaptor — the small metal box included with the interface. Attach the RF adaptor to the terminals on the TV set labeled VHF (not UHF). Then plug the video cable (also supplied with the interface) into the RF adaptor and into the jack labeled TV in the back of the interface.

Turn on the TV set and set it to either channel 3 or 4, depending on which is the weaker station in your area. Then set the switch on the back of the interface labeled Channel Select to this same channel. Finally, plug the disk/video interface power cord into the wall outlet, and you're ready (at last!) to go.

Starting Up the System

Now that everything is connected, getting the system running isn't hard. Turn on the Model 100 and *then* (not before) turn on the interface, using the power switch on the back panel. (The Model 100 should be in Menu Mode; that is, the main menu should appear on the screen.) The message "Please Wait!" will appear briefly on the TV screen followed by the message "Insert system diskette".

Inserting the System Diskette

In the following discussion, we'll assume that you have only one disk drive installed. If you have two, you can simply ignore the lower one for the time being.

The system diskette is a 5 1/4-inch floppy diskette, which you'll find in the back of your *TRS-80® Disk/Video Interface Owner's Manual*. It's important that you don't damage the information on this disk, because it's the only system disk you have at this point. One way to protect the information on a disk from accidental erasure is to put a *write protect tab* on it. This is just a small sticky label, folded over the "write protect notch" on the diskette. This notch is located on the side of the diskette, near the label; once it's covered,

you can't write new information on the disk. You should protect the system disk in this way before going any further.

Now, place the system diskette in the disk-drive slot so that the label is toward *you* (not the interface) and facing upward (see Figure 12-4).

Gently push the diskette all the way in. Now, close the lever on the side of the disk drive by turning it from its sideways position to a downward position. As soon as you do this, you'll see the message (very briefly) "Now, getting ready . . .", which will be followed by this screen display (again, on the TV screen, not on the Model 100):

```
COPR, 1983, MICROSOFT, CORP,  
ALL RIGHTS RESERVED  
LICENSED TO TANDY CORPORATION  
VERSION 01.00.00
```

If some other message appears on the TV screen, you may have used a diskette that is not a system diskette, or there may be some other malfunction. Check the diskette; if it's the right one, contact your dealer — you've got trouble.

The Model 100 will still be in Menu Mode, waiting for your commands.

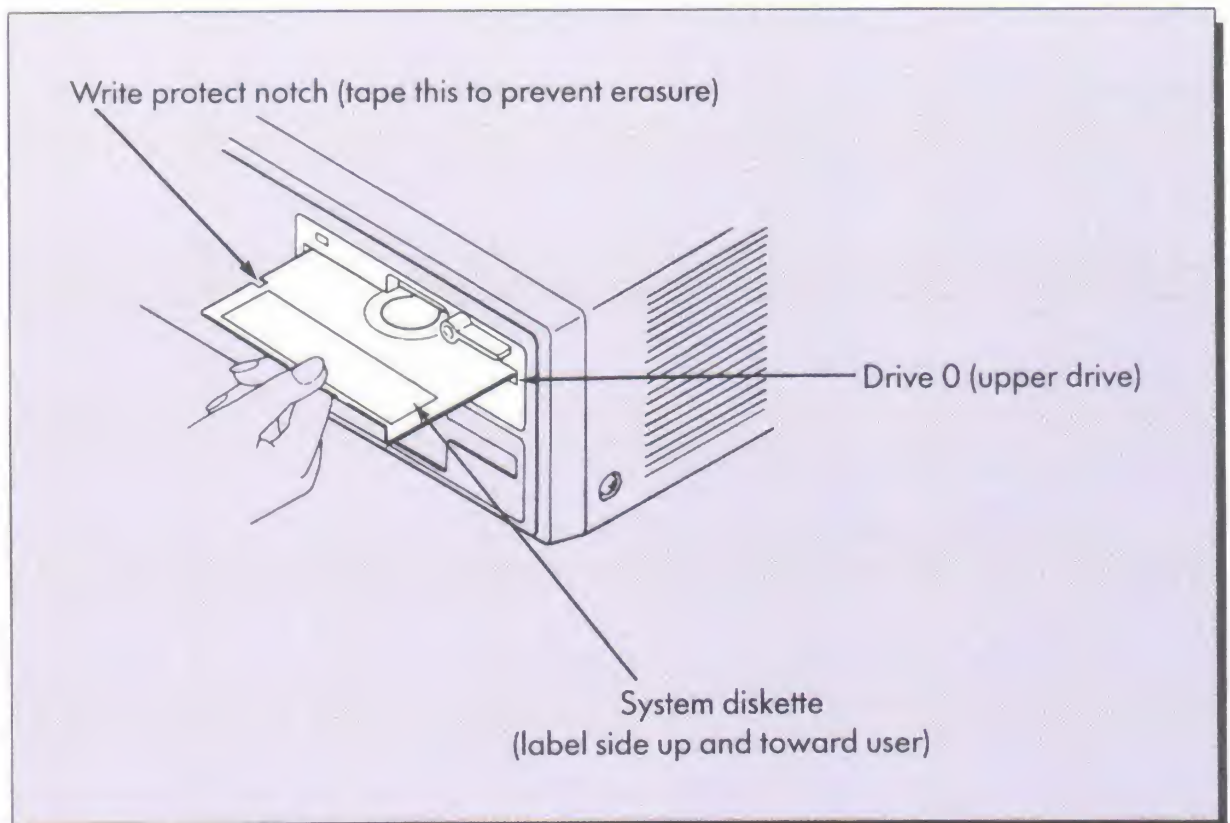


Figure 12-4. Inserting the system diskette

What's Changed in the Interface?

Although you can't tell by looking, you've now loaded the *disk-operating system (DOS)* from the system diskette into the memory of the disk/video interface. The DOS contains the instructions necessary for the interface to operate the disk system, which is why you need the system diskette to get the system started. (In the next section, you'll see that instructions must be loaded into the memory of the Model 100 as well.)

A Warning

Before we go any further, you should remember a simple warning: *when you turn off the disk/video interface, the Model 100 must be in Menu Mode*. You don't need to know this quite yet, but don't forget it when you go to shut down your system.

Getting Organized

We could have called this section "Loading Disk BASIC", but that's a little misleading. Here we're simply trying to put the Model 100 and the interface into a state where we can use them; however, the instructions we give to the disk system *are BASIC instructions*. Not only that, they are BASIC instructions that don't exist in the BASIC built into the Model 100; they are special instructions that must be *added to* the Model 100's BASIC.

Where do these additional BASIC instructions come from? They're stored on the system diskette, just like DOS, and we follow a procedure that causes them to be loaded automatically into the Model 100's memory. Here's how to do it.

The Model 100 screen should be displaying the main menu, and the TV screen should be displaying its original sign-on message:

```
COPR. 1983, MICROSOFT, CORP.  
ALL RIGHTS RESERVED  
LICENSED TO TANDY CORPORATION  
VERSION 01.00.00
```

On the Model 100 screen, position the cursor over the word *BASIC* and press **ENTER**. You should get the regular BASIC sign-on messages, so the screen should look like this:



To show you what we'll accomplish in the next few steps, we're going to perform a simple "before and after" experiment. Here's the "before" part: type in the BASIC direct command *LFILES*. You should get an error message, like this:

```
TRS-80 Model 100 Software
Copr. 1983 Microsoft
29382 Bytes free
OK
LFILES
?FC Error
OK
```

FC means "illegal function call", so BASIC at this point doesn't know what you mean by *LFILES*.

Now, turn the Model 100 (*not* the interface) *off and then on again*, using the regular power switch on the right side of the case. You should hear the disk drive whirring in the interface and see its little red light glow. The Model 100 screen will blank out briefly, and then the main menu will appear. The TV screen will blank out, and the message that was there before will be replaced by a flashing cursor.

Remember, you must be in BASIC, not the main menu, when you turn the switch on and off again; otherwise, the process won't work.

Memory Space

What happened when you switched the Model 100 off and then on? The interface magically transferred new instructions to add to the BASIC already built into the Model 100. You can tell that something like this must have happened, because the number of free bytes has changed: on our 32K Model 100, with no files in memory other than the built-in ones, the “Bytes free” dropped from 29382 to 24685 — almost 4,500 bytes. Yours will drop a similar amount, even if you started with less memory.

This brings up an important point: before you perform the above steps, you must be sure that there are at least 5,000 free bytes (preferably much more, so you’ll have space to work) available in the Model 100. If there aren’t, you’d better save some of your files to cassette tape and “kill” them in the Model 100’s memory.

Let’s get back to our “before and after” experiment. Again position the cursor over the word *BASIC* and press **ENTER**. The sign-on message will look normal, except that there will be fewer bytes free. Now type *LFILES* again. This time, instead of an error message, you should see a display of all the files on the system diskette:

```
TRS-80 Model 100 Software
Copr. 1983 Microsoft
24685 Bytes free
OK
LFILES
SYSTEM VER 01.00.00,
FORMAT,      1      BACKUP,      2
BACKUP.SNG 2
155.25 K AVAILABLE
OK
```

This tells you the version number of the system and that there are three files on the system diskette: *FORMAT*, *BACKUP*, and *BACKUP.SNG*. It also tells you how much space is left on the system diskette for your own programs: 155.25K, or somewhat more than 155,000 bytes.

Switching Screens

Using *LFILES* shows that you can use the Model 100 screen to control the disk drive. However, you can now use a larger screen, so why not take advantage of it? To switch to the larger screen, simply issue the following BASIC command:

```
SCREEN 1,1
```

Now you get the BASIC Ok prompt on the TV screen. The bottom of the screen will display the usual labels found in BASIC: File, Load, Save, Run, List, and (over on the right) Menu. Now, try entering the command LFILES again. What you type will appear on the TV screen, and the disk files will also be listed on the TV screen.

It's a lot easier to read on that big bright screen, isn't it? Also, notice how much more room there is on the screen: 25 lines instead of 8. This will come in handy when you start using the disk/video interface system with TEXT or BASIC.

Your system is now ready to go. You've loaded DOS into the memory of the disk/video interface and extra BASIC instructions into the memory of the Model 100. The screen output now appears on the TV screen. Now you can start using the system to accomplish useful tasks.

Backing Up the System Diskette

Now that your system is running and ready to go, you should make a copy of your system diskette. Later in this chapter, we'll be writing files to the system diskette, and if something should go wrong (remember Murphy's law), it would be embarrassing if your only system diskette were destroyed. Also, you'll eventually want to use lots of different diskettes, so it's a good idea to know how to prepare them.

Backing up the system diskette consists of several steps. First, you need to format a blank diskette, then you need to transfer the information from the original system diskette to the new one.

Formatting a Blank Diskette

You should now be looking at the TV screen — anything you type on the Model 100 keyboard will appear on this screen. Enter the following BASIC command:

```
RUN"0:FORMAT"
```

This instruction tells your augmented version of BASIC that you want to RUN a program called FORMAT, which is located on disk drive 0. (If you have only one drive, it has the number 0; if you have two drives, the upper one will be number 0, and the lower one will be number 1.) Note that if you try to use this instruction with the old BASIC, it won't work; you need to add the new instructions for it to recognize this format.

The screen will now display the following message:

```
This utility formats diskettes.  
- All data will be lost -
```

```
Which drive will be used (0 or 1)?
```

Enter the number 0, because you'll be using disk drive 0. The screen will display the following message:

```
Put the diskette to be formatted in  
Drive 0  
Press ENTER when ready.
```

Be careful here. You don't want to format your only system diskette, because formatting destroys everything on the diskette. *So be sure to remove it at this point.* Then insert a blank disk, either one fresh out of the box or one whose contents you don't mind destroying. Note that the disk to be formatted should *not* have the "write protect tab" on it.

Press **ENTER**. The disk drive will whir, and the message "Formatting!" will flash on the screen. It will take about thirty seconds to format the diskette. When the process is over, the message

```
FORMAT COMPLETE  
number of flawed tracks: 0
```

```
Ok
```

will appear on the screen, the Ok indicating that you're back in BASIC. The new disk has now been formatted.

Copying the System Diskette onto the Blank Diskette

To copy the contents of one diskette onto another, we'll use the program called BACKUP.SNG, which is already on the system diskette. Make sure that the system diskette is in drive 0; that is, remove the diskette you just formatted and reinsert the system diskette (with the "write protect notch" covered).

From BASIC, enter the following:

```
RUN"0:BACKUP.SNG"
```

You'll see the following message:

```
SINGLE DRIVE BACKUP UTILITY  
  
COPY ALL (Entire Diskette) or  
SYSTEM (System Files Only)  
  
Enter A (ALL) or S (SYSTEM)?
```

Enter an A, because you want an exact copy of the diskette.

Now you will be given a series of commands, telling you which diskette (formatted or source) to insert. Follow these instructions carefully until the message COPY COMPLETED appears. Your new diskette is now an exact copy of the original system diskette. Use it from now on and put the original away for safekeeping.

Using TEXT with the Disk/Video Interface

There are two main uses for the disk system — storing BASIC programs and storing TEXT files. (You can also use BASIC commands to write data to the disk, but that is beyond the scope of this book. For more on using BASIC with the disk/video interface, see *Mastering BASIC on the TRS-80® Model 100*, by Bernd Enders [New York: Plume/Waite, New American Library, 1984].)

In this section, we'll show you how to use the disk system with the TEXT program; in the next section, we'll explain how to store BASIC programs on the disk system.

Editing on the TV Screen

You should still be in BASIC, with the display appearing on the TV screen. If so, press function key **(F8)** to return to the menu on the Model 100 screen. (If you're already in Menu Mode, you won't need to do this.) The TV screen will freeze, and you'll find that you can move the cursor around the main menu. Select TEXT and press **(ENTER)**.

When you're asked "File to edit?", you can either make up a new name or enter the name of a file that already exists in the Model 100 main menu. When you enter the filename, you'll be treated to a new experience: the TEXT program will appear on the big TV screen! This is a great convenience. You can see the text more clearly, and you have 25 lines of display rather than eight. Otherwise, all the editing functions of TEXT operate just as described in Chapter 3. For instance, to see what the function keys do, press the **(LABEL)** key.

If you opened an existing text file, you'll have some prose on the screen. If you started a new text file, type something in so that you'll have a file to experiment with.

Saving a TEXT File to Disk

Here's how to save the text file to the disk. First, press function key **(F3)**, and the message "Save to:" will appear on the bottom of the TV screen. Then type a zero, followed by a colon and the name you want your program to have on the disk (not necessarily the name it has in the Model 100's memory). Don't forget the zero and colon. If you do, the Model 100 will think you want to save something on the cassette, and there will be a long pause while it sends signals out to the cassette deck, even if none is attached. If the file is long, this pause will be correspondingly long. If you get locked up this way, you can escape by typing **(SHIFT)** **(BREAK)**.

```
Save to: 0:LETTER
```

Want to see if the text file is really saved on diskette? Exit TEXT by pressing **(F8)**, which will take you back to the main menu on the Model 100 screen. Now, move the cursor to BASIC and press **(ENTER)**. Then issue the BASIC command "LFILES." All the files on your diskette will be listed on the TV screen, including the new text file, LETTER, which you just created.

Loading a Text File from Disk into the Model 100

Suppose you've saved a text file on the disk, and you want to do some more work on it using TEXT. How do you move it from the diskette into the Model 100?

If you're in BASIC, return to the main menu with **(F8)**. Select TEXT, and answer the "File to edit?" question with any name you like. Now you're in TEXT on the TV screen. Press function key **(F2)**, and the message "Load from:" will appear on the bottom of the screen. Enter the name of the program on the diskette.

```
Load from: 0:LETTER
```

Again, don't forget the zero and colon, or you'll be stuck waiting endlessly for the Model 100 to find the (probably) nonexistent program on the cassette drive. If this happens, type **(SHIFT)** **(BREAK)**.

If you typed the filename correctly, the file will appear almost instantly on the TV screen, ready for you to edit, print out, or save it to the Model 100 RAM, or whatever you want.

Using BASIC with the Disk/Video Interface

You've already learned how to get into BASIC and organize things so that BASIC commands and programs appear on the TV screen rather than the Model 100 screen. (Follow the steps described earlier if you've forgotten how to do this.) Now we'll show you how to save BASIC programs to disk and load them from disk back into the Model 100's memory.

We'll assume that you're already in BASIC. If not, return to the main menu and get into BASIC in the usual way.

Saving Programs to Disk

In order to save a program to disk, you need a program to save. You can either load one from the Model 100's memory (as described in Chapter 6), or you can type one in. Either way, the program is sitting in the Model 100's "active" BASIC workspace. To save the program to disk, press function key **(F3)**, which will print the message "Save" on your screen. Then type the zero and colon, followed by the name you want your program to have on the disk.

```
Save "0:ZAP"
```

If you want to verify that the program has been saved, simply type in the BASIC command **LFILES**. You'll see the program listed along with the other files on the disk.

Loading Programs from Disk

Loading a program from the disk into BASIC is equally straightforward. From BASIC, press function key **(F2)**, then type the zero and colon, followed by the name the program has on the disk. (Check it with **LFILES** if you've forgotten.)

```
Load "0:ZAP"
```

Something to remember here is that the filenames on the diskette are not given extensions, such as ".BA" and ".DO", as they are in the Model 100's memory. You can try to load a file that is not a BASIC program into BASIC, but, if you do, you'll get an "FF Error", which is "direct statement in file". This is what BASIC tells you when it tries to read text as a BASIC program.

Summary

In this chapter, you've learned how to use the optional disk/video interface unit with your Model 100. This unit gives your Model 100 vastly increased capabilities, though it does so at the expense of portability.



13

The Bar Code Reader — More Versatile than You Think

Concepts

- Uses for the bar code reader
- Three kinds of bar codes
- Setting up the bar code reader
- Setting up the machine language software
- Operating the BASIC software
- Operating the inventory programs

*I*n the preceding chapters you have learned how to input information from the outside world to the Model 100 by using the keyboard, a cassette tape, another computer, the phone lines, or the disk drive. This may seem like a wide enough variety of ways for the Model 100 to digest data, but there is another way — the bar code reader. Why is this additional input device necessary, and what can you do with it?

What Are Bar Codes?

Unless you live in Tierra del Fuego, you probably already possess many objects that have bar codes printed on them. Almost all canned, bottled, or packaged food products have bar codes on them, as do magazines, paperback books, and many hardware products. Wholesalers and stores use these codes to quickly and automatically record the sale of their goods; a computer

automatically calculates each item's price and simultaneously updates the store's inventory to reflect the sale.

Typical bar codes on consumer products, or uniform product codes (UPC), are shown in Figure 13-1.

The use of bar codes goes well beyond cans of beans and paperback books, however. The Department of Defense uses them to keep track of spare parts on aircraft carriers, and hospitals use them to label patients as well as such items as containers of blood. Runners in the Boston Marathon wear them on their jerseys so they can be scored electronically, and salesmen in various fields use catalogues with bar codes printed next to the product description so that orders can be automatically entered into a computer. There is even a company that markets children's books with bar codes printed in them. The books are sold with a bar code reader and a small computer. When the child reads the bar code next to a picture of a bear, for example, the computer generates the sound of a bear.

There are actually several different kinds of bar codes. The one used on retail products is called a uniform product code (UPC). It is a rather complex code, with a sophisticated error-checking system built in to ensure that



Figure 13-1. Typical UPC codes

no errors are made when the code is read. Two other kinds of codes are in common use — Code 3 of 9 and Plessey. Code 3 of 9 is mostly used in inventory control. It is much simpler to understand than UPC — in fact, with a little practice, you can read it yourself (though you may need a magnifying glass). Plessey is also used in inventory control systems.

The Model 100 Bar Code Reader can, in concert with the software that comes with it, read all three of these major bar code systems.

What's So Great About a Bar Code Reader?

You may think that the bar code reader is useful only in industrial situations such as inventory control, but it is actually a surprisingly useful and versatile device; its potential uses are limited only by your imagination. Its usefulness stems from the fact that it can read information quickly and accurately directly from the printed page. You needn't record the information on special media, such as cassette tapes or diskettes; further, you don't need to type it in by hand, with all the resultant error possibilities. Once you have the code printed on paper, it's easy to copy it; a photocopy machine is all you need.

Uses for the Bar Code Reader

The most obvious uses of the bar code reader are commercial. The Model 100 permits you to move freely around a warehouse or industrial site, reading bar codes and storing the information for analysis or transfer to another computer. As we've mentioned, the bar code reader is used widely in retail stores, but there are also a number of noncommercial applications.

You can read BASIC program listings directly from the pages of a magazine into the computer. Some magazines now publish programs listed in bar code format, and the popularity of this approach will undoubtedly increase in the future.

You could use the Model 100 to actually read and verify the prices and total bill on products you are buying. Though this might not be very practical at the supermarket, if you were buying a large number of hardware items, it might enable you to quickly and easily verify the accuracy of your purchase and of the bill.

If you have any sort of collection — stamps, coins, fossils, butterflies, flintlock rifles — you can use bar codes to mark all the items in your collection. Then you can use your Model 100 to read the number of a particular item, and, using the ADDRSS program or a more sophisticated

database program, you can instantly recall all the relevant information about the item. In the case of a stamp, for example, this might be the country of origin, date, denomination, purchase price, and so forth.

One potential problem in using the bar code reader in a noncommercial application is the difficulty of generating the printed version of the code. The 3 of 9 code is probably the best bet for this because it's so easily understandable. Once you have examples of the codes for all the characters, you can make up any code or message you want by simply pasting together the codes for the individual characters. (The instruction book that comes with the bar code reader describes the 3 of 9 code in detail.) Of course, if your budget can support it, you can also buy machines that print out bar codes directly.

What You Need

The bar code reader is available from Radio Shack as an accessory for the Model 100. It consists of the wand, an instruction book, and a cassette containing the software necessary to use the wand.

The Wand

The wand is like a thick pencil. It contains a tiny light and a photo cell in its tip, and a built-in cable that plugs into the bar code reader socket on the left-hand side of the Model 100, as shown in Figure 13-2.

A switch is located on the side of the wand to turn on its light. Because the wand requires considerable power (at least compared with the computer itself), it should not be kept on continuously if you are running the Model 100 on batteries. For extensive use of the wand, Radio Shack suggests that you power the Model 100 with the optional AC adaptor.

The Software

There are two kinds of programs included on the cassette that comes with the bar code reader.

First, there are *machine language* programs. The wand must be able to quickly “see” and analyze the lines and spaces of the bar code as it is pulled across them. BASIC is too slow a language for this process — only a machine language program can operate fast enough. The cassette contains three of these programs, one for each of the three kinds of bar codes: UPC, 3 of 9, and Plessey.

There are also three *BASIC programs* on the cassette. These are used with the machine language programs; that is, both a BASIC program and

a machine language program must be loaded into the Model 100 simultaneously for the bar code reader to work. First, the machine language program obtains the numbers from the wand, then the BASIC program takes over and actually processes the number. This BASIC processing can be as simple as printing the number out on the screen or as complicated as entering the bar code into an inventory program. Figure 13-3 shows the relationship between the different components of the system.

The BASIC programs on the cassette are READBC.BA, which is a simple program to read bar codes and display them on the screen, and INVCRE.BA and INVENT.BA, which are used for creating and using an inventory file system.

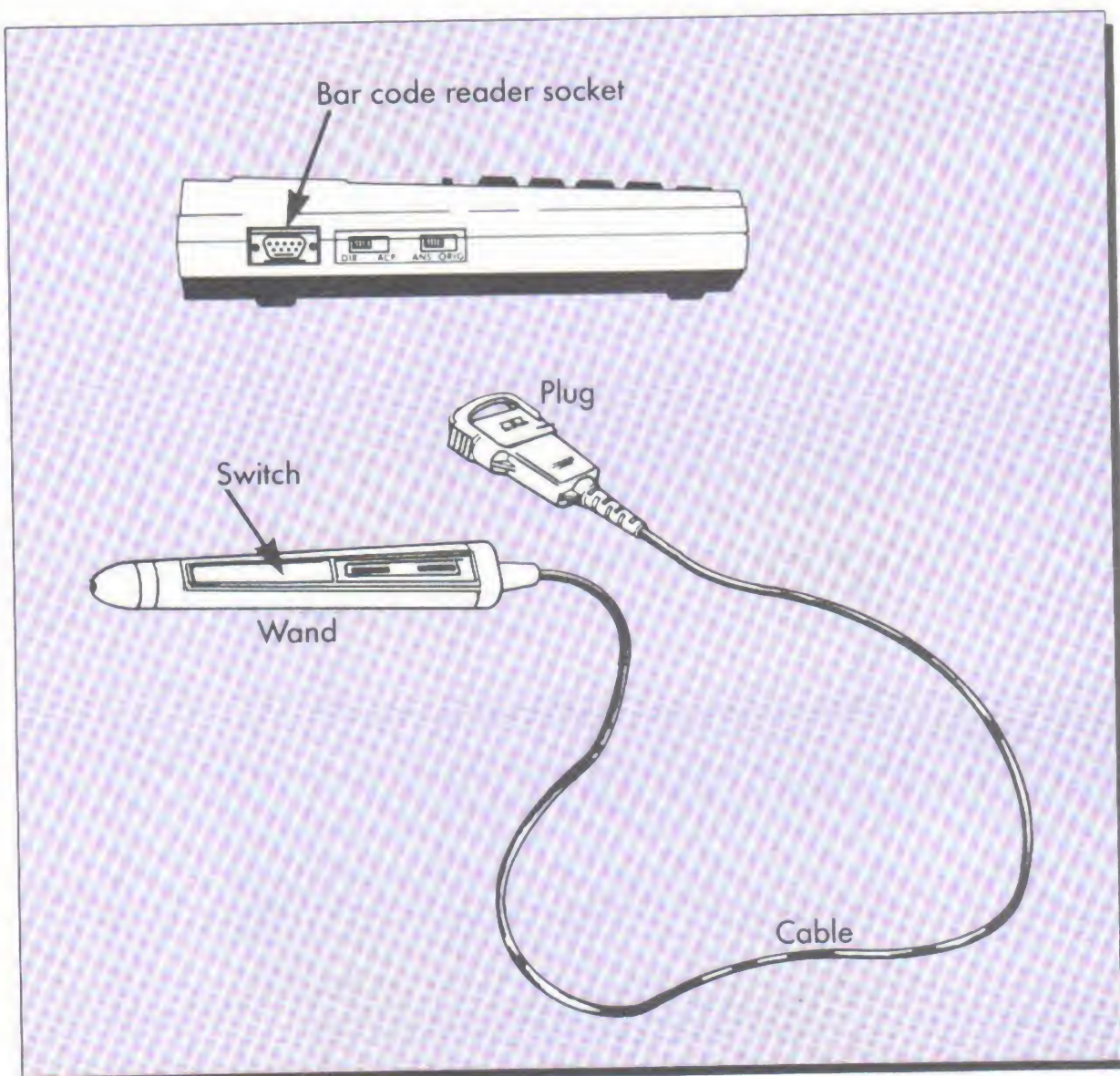


Figure 13-2. Wand, cable, and Model 100

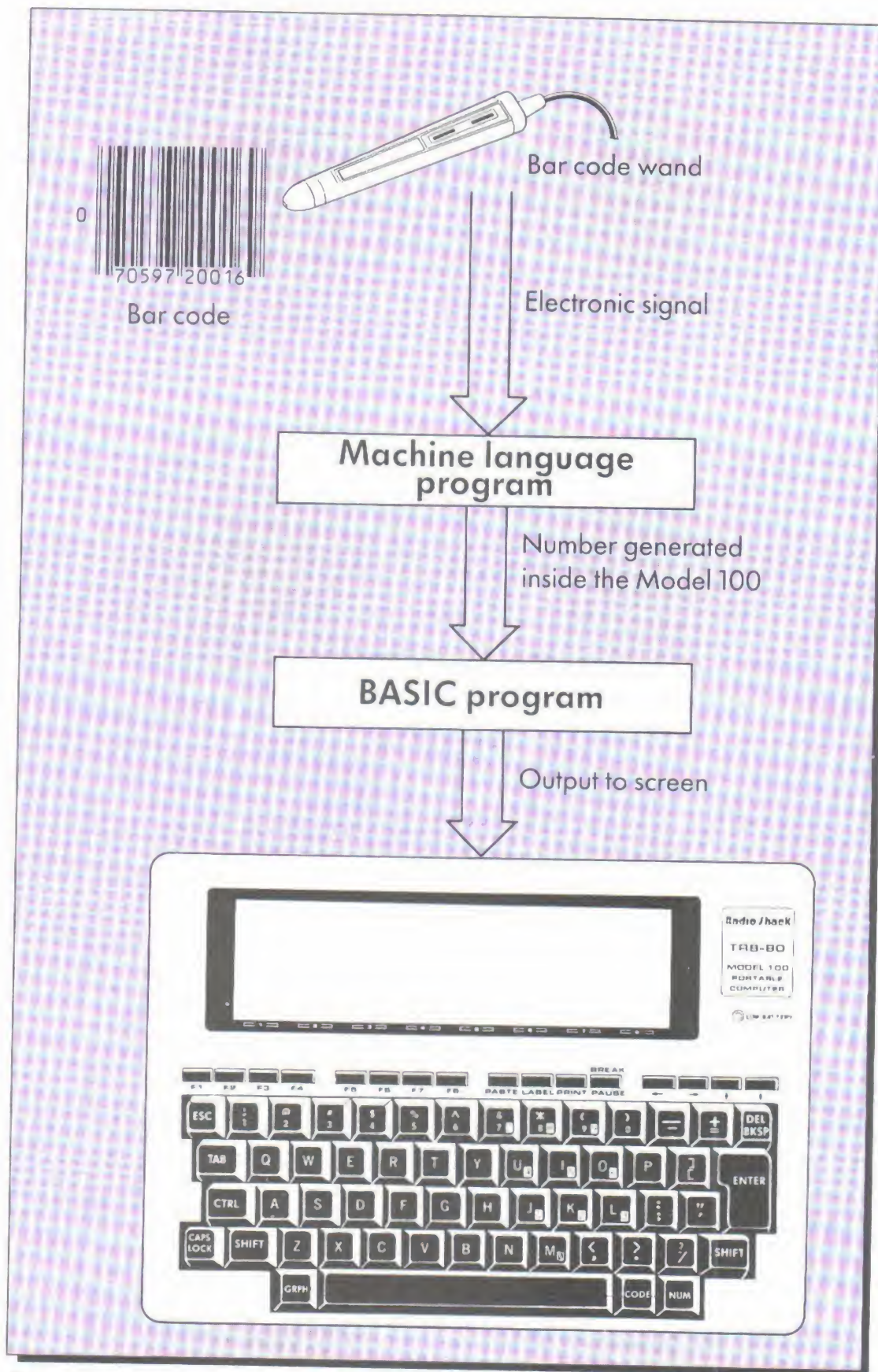


Figure 13-3. Bar code system components

Setting Up Your Bar Code Reader

It's not hard to start using your bar code reader. The physical connections are no harder than plugging in a toaster. Getting the software organized and operating is somewhat more complex but still much simpler than, say, using the disk drive.

Setting Up the Hardware

Here's the easy part. Simply take the connector attached to the cable on the end of the wand and plug it into the bar code reader socket on the left side of the computer. This socket is located next to the two switches marked DIR-ACP and ANS-ORIG. You can't put the plug in backward, due to the shape of the socket. Don't try to do anything with the wand yet; its special software must be loaded and operating before the computer can understand what the wand is telling it.

Setting Up the Machine Language Programs

Organizing the software is somewhat complicated, partly because it involves using the cassette recorder. Refer to Chapter 7 to review setting up a cassette recorder and loading BASIC programs from the cassette into the Model 100.

Loading Machine Language Programs

Though we explained in Chapter 7 how to load BASIC programs from cassette, you'll also need to know how to load machine language programs. Here's how that's done.

Set Up the Recorder

First, set up the cassette recorder just as you would for a BASIC program: cables plugged in correctly, volume set to 5, tape rewound to the beginning of the cassette, and Play button on the recorder depressed.

Make Sure You Have Enough RAM

The three machine language programs on the cassette each occupy about 830 bytes of memory, READBC.BA uses 430 bytes, and the two inventory programs (INVCRE.BA and INVENT.BA) each use about 1,100 bytes. In addition, the inventory programs will generate a document file called INVTRY.DO (more on this later). The length of this file depends on how much information you put into it, but let's assume you'll need about 1,000 bytes for it. When you're experimenting with the bar code reader, it's

simplest to load all these programs into RAM at the same time; so you should have somewhat more than 6,000 bytes of free memory before you begin.

Organize Your Memory Space

The next thing you must know is that machine language programs are not loaded into memory in the same way that BASIC programs are. Because both the machine language program and the BASIC program will be in active memory at the same time, you must take care that BASIC knows where the machine language program is, so that it will not write over it. This is accomplished with a CLEAR command, which must be given *before* any machine language programs are loaded. For the particular machine language programs to be used with the bar code reader, you'll need the following command:

```
TRS-80 Model 100 Software  
Copr. 1983 Microsoft  
8143 Bytes free  
Ok
```

```
clear 110, 61788
```

← You type this

The first number in the CLEAR statement sets aside 110 bytes of storage for the string variables that will be read in by the machine language program. The second number tells BASIC not to write anything above address number 61788; the space above this is reserved for the machine language programs. To issue this command, you must be in BASIC. (Review Chapter 6 if you're not clear on how to get into BASIC or issue direct commands.)

Issue BASIC Loading Commands

The three machine language programs are arranged on the cassette tape in the following order: B3OF9, UPC, PLESY. We'll load them in this same order to avoid having to rewind the tape.

To load B3OF9, type the following direct command from BASIC:

```
cloadm"B3OF9"
```

Notice the *m* following the *cload*. This tells BASIC that you're loading a machine language program rather than a BASIC program. When you press **ENTER** following this command, the tape will begin to turn and the program will be loaded into the Model 100.

As it loads the program, BASIC will display three important addresses: the lowest memory address occupied by the program, the highest memory address, and the address where execution will begin. You needn't know anything about what these addresses do, but you do need to know what they are. The screen should display the following:



Save the Program in RAM

Now the B30F9 program is in active memory, but you need to load it into permanent RAM storage. To do this, you must issue some more special commands, which make use of the address already printed out. These are the following:

```
savem "B30F9", 61788, 62611, 61824
```

Notice that these are the same numbers, in the same order, as displayed earlier.

The program should now be saved in memory, which you can verify by pressing **(F8)** to escape from BASIC to the main menu. There you should find B30F9.CO as one of the files listed. The ".CO" is the extension used for machine language programs.

Repeat Process for UPC and PLESY

Now you should return to BASIC and follow this same procedure for the remaining two programs, UPC and PLESY. The only differences in the procedure are the name of the program and the specific addresses used.

Setting Up the BASIC Programs

Loading the BASIC programs into memory should be easier than loading the machine language programs, because you've already become familiar with the process in Chapters 6 and 7. There are three BASIC programs — READBC.BA, INVCRE.BA, and INVENT.BA. To load READBC.BA,

make sure the Play button on the recorder is depressed. The tape should still be stopped where it was following loading of the machine language programs. Now type

```
load"READBC"
```

The program will load, and the screen will display the following:

```
Found:READBC
```

To save this program in RAM, you type

```
save "READBC"
```

When you exit from BASIC with **(F8)**, you'll find this program listed on the main menu as READBC.BA.

Follow this same loading process with the two BASIC inventory programs INVCRE.BA and INVENT.BA.

Your system is now set up and ready to operate.

Operating Your Bar Code Reader System

The software provided with your bar code reader provides two levels of complexity. The program READBC lets you experiment by simply reading bar codes and displaying the resulting numbers on the screen. The inventory programs INVCRE and INVENT let you create an actual inventory file that lists items, their bar codes, and the quantity of each item.

Using the READBC Program

Let's start out with the simplest application program, READBC. From the main menu, position the cursor over the file READBC.BA and press **(ENTER)**. The screen will display the following:



The uniform product code is the most commonly available code, so let's choose that one by typing 1. As soon as you press the numeral, the screen will clear and you'll see the following message:

Wand Ready:

This is the moment you've been waiting for. Will the wand really work?

Operating the Wand

Using the wand is easy in principle, but it demands a certain finesse. Check that the wand is properly plugged in. You can make certain that it's operating by looking into the pointed end of the wand and pressing the switch on the side. You should see a dim red light shining out from the little hole in the end of the wand.

Now, hold the wand as you would a pencil and move it rapidly across a bar code. (The codes on magazines are easy to experiment with because they are on a nice flat surface.) The wand should be held at about a 20-degree angle to the surface of the paper, though this isn't critical. Straight up and down works, too. A common mistake is to move the wand too slowly. If you go too slow, nothing will happen. Sometimes it works better to read from right to left, though left to right is more natural.

You may need to experiment to make the wand work. It seems to be easy for some people to get the hang of it, while others have some trouble. If you're having trouble, get a friend to try it.

If all goes well, the Model 100 will beep, and the number corresponding to the bar code will be displayed on the screen:

Wand Ready:07148602601

Wand Ready:

That's all this program does: you read the bar code, and the program displays it on the screen. It's a useful program to experiment with, but its practical applications don't go very far. To escape from the program, press the **ENTER** key. You'll be asked if you want to resume reading from the wand; type Y (for "yes") or N (for "no") as appropriate.

Don't try to break out of the program with the **SHIFT** **BREAK** key combination. It's important that any program that uses the bar code machine language routines terminate normally; otherwise, the computer will be left in a nonstandard state, and your other programs may not behave normally.

Using the Inventory Programs

The inventory programs INVCRE and INVENT are considerably more sophisticated than READBC. INVCRE lets you *create* an inventory file that contains descriptions of items and their bar codes, which you read in with the wand. The file it creates is a document file called INVTRY. The INVENT program lets you *use* the INVTRY file to record the quantity of each item.

Creating the INVTRY File with INVCRE

From the main menu, position the cursor over the file INVCRE.BA and press **(ENTER)**. You should see the following screen:



Type 1, because you want to create an inventory file. You will be greeted with another screen, which warns you that you will erase any existing inventory file. Type C (or c) to continue. You will be told:

Scan Item Now

Use the wand to scan the bar code on the first item. When the number has been successfully input to the program, a new message will appear:

Description (Type end to quit)

You respond to this by typing the description of the item. This should be less than 40 characters. The resulting screen might look like this:

```
Build Inventory File
Scan Item Now
01402414355
Description (Type end to quit)
Totable Computer Magazine
```

As soon as you press **ENTER**, this item and its number are entered in the INVTRY file. The program then redisplay the screen, asking you to

Scan Item Now

Keep scanning different items and entering their descriptions until you run out of items.

To terminate the program, scan any bar code (it will be forgotten by the program); then, when you are asked for the description, enter the word *end*. You will be returned to the INVCRE menu. To exit from the program, type 3 to quit.

Updating the INVTRY File with INVENT

Now that you have created the file containing the list of items you want to inventory, along with their codes, you use the INVENT program to actually enter the quantity of each item that is on hand.

From the main menu, position the cursor over the file INVENT.BA and press **ENTER**. The screen should display the following:

```
Simple Inventory System
Press  Action
1...Take Inventory
2...Print Inventory Listing
3...Quit
```

Type 1 to take the inventory. The screen will display the following:

Scan Item Now

Use the wand to scan the bar code of the item whose quantity you want to record. If you scan a code that has not previously been recorded in the INVTRY file, the Model 100 will emit a low-pitched beep and display the following message:

No Match Found

If the bar code matches one already in the file, the screen will display the description of the item (which you entered with INVCRE) and ask you how many of this particular item there are. The resulting screen looks like this:



When you're finished, you exit from the program by scanning any bar code and then entering E (or e) instead of the quantity. This will return you to the INVENT menu. To exit from INVENT back to BASIC, type 3 to quit.

Printing the INVTRY File

To obtain a hard-copy printout of your inventory file, you can use option number 2 in the INVENT program. To use this option, you must have a printer hooked up to your Model 100. (For a description of how to do this, see Chapter 11.)

From the INVENT menu, type 2. The program will ask if your printer is ready; you type any key to continue. Your INVTRY file will then be printed out on your printer. It will look something like this:

Simple Inventory System Listing

Item	Description	Quantity
01402414044	Totable Computer Magazine	100
02705414144	Widgets Digest	144
07409414074	Small Computer Bulletin	1048

This system of inventory programs is straightforward and works remarkably well for a simple application. For more sophisticated applications, you will need to either buy a more complex program or have one custom written for you.



BASIC Error Messages

If you try to run a BASIC program which contains an error, or if you issue a direct command in BASIC (like "KILL", or "LOAD") which contains an error, the Model 100 will respond with an error message. It will look something like this:

?SN Error

By consulting this appendix, you can get some clue as to the nature of the error. The most common error is the one shown above: a Syntax Error. This means that the BASIC interpreter couldn't understand the line you typed. This is usually due to a typing error. Other errors are somewhat more obscure; unraveling them may require some knowledge of BASIC.

Message	Meaning
AO	Already open
AT	Bad allocation table*
BN	Bad file number
BS	Bad subscript
CF	File not open
CN	Can't continue
DD	Doubly dimensioned array
DF	Disk full*
DN	Bad drive number*
DS	Direct statement in file
EF	Input past end of file
FC	Illegal function call
FE	File already exists*
FF	File not found
FL	Undefined error
ID	Illegal direct
IE	Undefined error
IO	Error
LS	String too long
MO	Missing operand

NF	NEXT without FOR
NM	Bad file name
NR	No RESUME
OD	Out of data
OM	Out of memory
OS	Out of string space
OV	Overflow
RG	RETURN without GOSUB
RW	RESUME without error
SN	Syntax error
ST	String formula too complex
TM	Type mismatch
TS	Bad track/sector*
UE	Undefined error
UL	Undefined line
/O	Division by zero

*Used only in *disk*-BASIC



ASCII Character Codes

Decimal Value	Printed Character	Keyboard Character
32		SPACEBAR
33	!	!
34	"	"
35	#	#
36	\$	\$
	%	%
37		
38	&	&
39	,	,
40	((
41))
42	*	*
43	+	+
44	,	,
45	-	-
46	.	.
47	/	/
48	0	0
49	1	1
50	2	2
51	3	3
52	4	4
53	5	5
54	6	6
55	7	7
56	8	8
57	9	9
58	:	:
59	;	;
60	<	<
61	=	=
62	>	>

63	?	?
64	@	@
65	A	A
66	B	B
67	C	C
68	D	D
69	E	E
70	F	F
71	G	G
72	H	H
73	I	I
74	J	J
75	K	K
76	L	L
77	M	M
78	N	N
79	O	O
80	P	P
81	Q	Q
82	R	R
83	S	S
84	T	T
85	U	U
86	V	V
87	W	W
88	X	X
89	Y	Y
90	Z	Z
91	[[
92	\	GRPH —
93]]
94	^	^
95	—	—
96	\	GRPH [
97	a	A
98	b	B
99	c	C
100	d	D
101	e	E
102	f	F
103	g	G
104	h	H

105	i	I
106	j	J
107	k	K
108	l	L
109	m	M
110	n	N
111	o	O
112	p	P
113	q	Q
114	r	R
115	s	S
116	t	T
117	u	U
118	v	V
119	w	W
120	x	X
121	y	Y
122	z	Z
123	{	GRPH 9
124		GRPH —
125	}	GRPH 0
126	~	GRPH]
127		DEL

Index

- ADDRSS, 47-61, 82
- ADRS.DO files, 53
- Acoustic coupler, 101, 103
- Acoustic coupler hookup, 103
- Address file program, 58
- Address list, 47
- Auto-log-on procedure, 126
- Auto-log-on program, 125
- Autodial, 50, 52
- BASIC, 63-73, 156
 - commands, 68
 - editing rules, 71
- Bar code reader, 159-73
- Batteries, 15, 87
- Block of text, 30
- Bytes, 8
- Calendar, 21
- Cash flow, 44
- Cassette storage, 85
- Clock, 21
- Coding an auto-log-on
 - sequence, 125
- CompuServe, 115
- CompuServe commands, 120
- Computer terms, defined,
 - 7-11
- Creating a program, 64
- Cursor, 9
- Cursor movement, 27
- DIP switch, 77
- DIR, 100
- Database, 113
- Date, 19, 39
- Delete, 29
- Deleting a block, 30
- Deleting a file, 72
- Dialing from ADRS.DO, 107
- Direct commands, 64
- Disconnect, 111
- Disk space, 116
- Disk/video interface, 141-57
- Dow Jones, 125
- Dow Jones News/Retrieval (DJNR), 125
- Download, 136
- Downloading, 110
- EDIT mode, 72
- Editing a BASIC program, 71
- Editor, 23
- Electronic mail, 115, 124
- Encyclopedias, 115
- Envelope, 82
- Errors, 28
- FIND function, 34
- Filename, 19, 25
- Floppy disk, 141
- Free software, 116
- Function key(s), 68, 111
- Games, 116
- Hookup to a printer, 76
- Hookup to other computers,
 - 129-39
- IPL (Initial program load), 70
- Information, 116
- Information networks, 113-27
- Initial program load, 70
- Insert, 29
- KILL command, 73
- Keyboard, 17
- LLIST, 80
- LOADing a TEXT file from
 - cassette, 90, 92
- LOADing a BASIC file from
 - cassette, 95, 96
- LOADing and RUNning a
 - BASIC file, 96
- LPRINT, 81
- Line feeds, 77
- List, 66
- Listing a program, 67
- Load, 70
- Loading BASIC files, 138
- Loading a TEXT file, 90
- Loading a text file from disk,
 - 135
- Mailing label program, 55
- Memory, 8
- Memory space, 151
- Menu, 16
- Modem, 11, 99, 100
- Modem hookup, 101
- Monitor, *See* Video monitor
- NAME command, 72
- NEW, 66
- NOTE.DO, 37-38
- Notepad, 44
- Numbered instructions, 64
- Null modem adaptor, 130
- Online databases, 113
- PASTE key, 32
- PRINT key, 79
- Parallel port, 11
- Phone lines, 99
- Phone numbers, 48, 50
- Pixels, 16
- Power sources, 15
- Power supply, 87
- Printer, 76
- Printing BASIC programs,
 - 80-81
 - LLIST, 80
 - SAVE "LPT:", 81

- Printing files, 75-83
- Printing mailing labels, 60
- Printing the screen, 79
- Printing whole text files, 79
- Printing with ADDRSS, 82
- Printing with SCHEDL, 82
- Project planning, 43

- RAM, 8, 85
- RESET, 13, 50
- ROM, 8
- ROM module expansion
 - compartment, 13, 142
- RS-232 port, 129
- Random Access Memory, 8
- Read Only Memory, 8
- Receiving (downloading) a
 - file, 110
- Renaming a file, 72
- Replace, 35
- Running a program, 69

- SAVE, 66
- SCHEDL, 37-45, 82
 - FIND, 40
 - LFIND, 42
 - STAT function, 105, 132, 137
- Saving a text file, 29
- Saving a BASIC file in TEXT
 - format, 95
- Saving a BASIC file to
 - cassette, 93, 94
- Saving a text file to cassette,
 - 88, 90
- Saving and loading BASIC
 - files, 137
- Saving and loading text files,
 - 131
- Saving text files to disk, 133
- Screen, 16
- Search and replace, 35
- Search for text, 34
- Sending (uploading) a file,
 - 109
- Sending BASIC files, 137
- Serial port, 11, 129
- Setting the date and time, 19
- Special interest groups
 - (SIGS), 124
- Storing a program, 68

- Storing files on cassette,
 - 85-97
- String, 34

- TAB key, 39
- TELCOM, 50, 99-111
- TEXT, 23-35, 154
 - correcting errors, 28
 - cursor movement, 27
 - paste, 30
 - select, 30
- TV set, 142, 146
- Telecommunications, 99
- Terminal mode, 108
- Text editing, 23
- Time, 19, 39
- Time management, 37

- Uniform product code (UPC),
 - 160
- Upload, 134

- Video monitor, 142, 146

- Wand operation, 169



(0452)

Other PLUME/WAITE books on the TRS-80® Model 100:

- ☐ **Mastering BASIC on the TRS-80® Model 100, by Bernd Enders.** An exceptionally easy-to-follow introduction to the built-in programming language on the Model 100. Also serves as a comprehensive reference guide for the advanced user. Covers all Model 100 BASIC features including graphics, sound, and file-handling. With this book and the Model 100 you can learn BASIC anywhere! (255759—\$19.95)
- ☐ **Games and Utilities for the TRS-80® Model 100, by Ron Karr, Steven Olsen, and Robert Lafore.** A collection of powerful programs to enhance your Model 100. Enjoy fast-paced, exciting card games, arcade games, music, art, and learning games. Help yourself to practical utilities that let you count words in a text file, turn your Model 100 into a scientific calculator, show file sizes, and generally increase your Model 100's usefulness, and your own grasp of programming. (255775—\$16.95)
- ☐ **Practical Finance on the TRS-80® Model 100, by S. Venit and Diane Burns.** The perfect book for anyone using the Model 100 in business: investors, real estate brokers, managers. Contains short but powerful programs to perform production planning, and access financial and other information from CompuServe® and the Dow Jones News/Retrieval® service. (255767—\$15.95)
- ☐ **Hidden Powers of the TRS-80® Model 100, by Christopher L. Morgan.** This amazing book takes you deep inside the Model 100 to reveal for the first time how it really works. You'll learn about the amazing power buried in the ROM, and how to use this power in your own programs. You can print in reverse video, prevent any screen lines from scrolling, dial the telephone from BASIC, control external devices from the cassette port, and discover many other fascinating secrets hidden within your Model 100. (255783—\$19.95)

All prices higher in Canada.

To order, use the convenient coupon on the next page.

Other Plume/Waite books available from New American Library:

- ☐ **BASIC PRIMER for the IBM® PC and XT by Bernd Enders and Bob Petersen.** An exceptionally easy-to-follow entry into BASIC programming that also serves as a comprehensive reference guide for the advanced user. Includes thorough coverage of all IBM BASIC features: color graphics, sound, disk access, and floating point.
(254957 — \$16.95)
- ☐ **DOS PRIMER for the IBM® PC and XT by Mitchell Waite, John Angermeyer and Mark Noble.** An easy-to-understand guide to IBM's disk operating system, versions 1.1 and 2.0, which explains — from the ground up — what a DOS does and how to use it. Also covered are advanced topics such as the fixed disk, tree-structured directories, and redirection.
(254949 — \$14.95)
- ☐ **PASCAL PRIMER for the IBM® PC by Michael Pardee.** An authoritative guide to this important structured language. Using sound and graphics examples, this book takes the reader from simple concepts to advanced topics such as files, linked lists, compilands, pointers, and the heap.
(254965 — \$17.95)
- ☐ **ASSEMBLY LANGUAGE PRIMER for the IBM® PC and XT by Robert Lafore.** This unusual book teaches assembly language to the beginner. The author's unique approach, using DEBUG and DOS functions, gets the reader programming fast without the usual confusion and overhead found in most books on this fundamental subject. Covers sound, graphics, and disk access.
(254973 — \$21.75)
- ☐ **BLUEBOOK OF ASSEMBLY ROUTINES for the IBM® PC and XT by Christopher Morgan.** A collection of expertly written "cookbook" routines that can be plugged in and used in any BASIC, Pascal, or assembly language program. Included are graphics, sound, arithmetic conversions. Get the speed and power of assembly language in your program, even if you don't know the language!
(254981 — \$19.95)

Buy them at your local bookstore or use this convenient coupon for ordering.

NEW AMERICAN LIBRARY
P.O. Box 999, Bergenfield, New Jersey 07621

Please send me the PLUME BOOKS I have checked above. I am enclosing \$_____ (please add \$1.50 to this order to cover postage and handling). Send check or money order—no cash or C.O.D.'s. Prices and numbers are subject to change without notice.

Name_____

Address_____

City_____ State_____ Zip Code_____

Allow 4-6 weeks for delivery
This offer subject to withdrawal without notice.

INTRODUCING THE TRS-80[®] MODEL 100

From initial set-up to advanced applications this complete, easy-to-follow guide shows you exactly what you need to get started on your Model 100 portable computer. Perfect for beginners or seasoned owners, it shows you how to get the most from the Model 100's built-in software: how to use the word processing program TEXT for writing letters and articles, and the best way to use the organizing and scheduling programs ADDRSS and SCHEDL. Techniques for typing in, editing, and running BASIC programs are clearly described.

In addition, there are instructions for connecting the Model 100 to cassette recorders, to printers, and even to other computers for sharing data via the TELCOM program. With the Model 100 and a telephone, you can send text or sales information to remote locations. This is also the first book on the Model 100 to cover the optional disk drive and video interface.

If you own a Model 100, or are thinking about purchasing one, this is the first book you should own.

The Waite Group is a Sausalito, California, based producer of high-quality books on personal computing. Acknowledged as a leader in the industry, the Waite Group has written and produced over thirty titles, including such best sellers as Assembly Language Primer for the IBM PC & XT, Graphics Primer for the IBM PC, CP/M Primer, and Soul of CP/M. Internationally known and award winning, Waite Group books are distributed worldwide, and have been repackaged with the products of such major companies as Epson, Wang, Xerox, Tandy Radio-Shack, NCR and Exxon. Mr. Waite, President of the Waite Group, has been involved in the computer industry since 1976 when he bought his first Apple I computer from Steven Jobs.



51857 01595

ISBN 0-452-25574